

Щасливський навчально-виховний комплекс
Бориспільської районної ради Київської області

ВОЛОДИМИР СТЕПАНОВИЧ ОСТАПЕЦЬ

**ПРОБЛЕМИ
ШКІЛЬНОЇ ІНФОРМАТИКИ**

Щасливе
2014

*Рекомендовано Вченою радою Київського обласного інституту післядипломної освіти педагогічних кадрів до участі у П'ятій міжнародній виставці
«Сучасні заклади освіти – 2014».
(протокол №1 від 14.01.2014 року)*

Рецензенти: Федорчук В. А.,

завідувач навчально-методичним кабінетом
інформатики й обчислювальної техніки
Київського обласного інституту
післядипломної освіти педагогічних кадрів

Скляр І. В.,

викладач УФМЛі КПІ ім. Т. Шевченка,
Заслужений учитель України

**Остапець В.С. Проблеми шкільної інформатики : методичний посібник /
В.С. Остапець. – Щасливе, 2014. – 163 с.**

У силу своєї специфіки предмет шкільної інформатики протягом усього часу з моменту введення в навчальні плани постійно перебуває в процесі оновлень та змін. Але особливо концептуальні, практичні та методичні зміни в шкільній інформатиці відчутні тепер, коли стрімко модернізується комп'ютерна техніка, з'являються зовсім нові технології опрацювання інформації, стає доступним Інтернет, життя вимагає перенесення базового курсу інформатики із старших класів у середні і навіть початкові. В цих умовах учителі інформатики стикаються зі значними труднощами у викладанні предмету. Тому надзвичайно актуально стоять питання виявлення проблем сучасної шкільної інформатики та пошуків шляхів їх подолання.

Методичний посібник присвячений вивченню проблемних питань викладання інформатики у школі та пошукам шляхів їх подолання. Розпочато цю роботу ще в 2007-2010 роках, коли працювала обласна школа передового педагогічного досвіду (ОШППД) при Київському обласному інституті післядипломної освіти педагогічних кадрів (КОПОПК, м. Біла Церква). Тоді були сформульовані і вивчені сім базових проблем. Протягом 2011-2013 років було продовжено вивчення цих проблем, але в умовах і на перспективу перенесення базового курсу інформатики в 5-9 класи. У посібнику досить детально проаналізовано сучасні навчальні програми та підручники, планування роботи учителя інформатики, роль інноваційних технологій навчання, надано поради учителям щодо методичної роботи проведення уроків та позакласної роботи з інформатики, запропоновано сучасні розробки уроків та практичних занять, надано методичні поради щодо виготовлення саморобних електронних дидактичних засобів навчання тощо. Місцями увага вчителів інформатики виноситься за межі навчального предмету, вивчаються питання якості навчання і освіти в цілому, сучасної мотивації навчання, місця шкільного курсу інформатики в загальній середній освіті.

Методичний посібник може бути корисним у першу чергу вчителям інформатики. Але його можуть використовувати також методисти з інформатики районних та обласних ланок і інші педагогічні працівники.

ЗМІСТ:

ВСТУП	4
Розділ I	
I.1 Актуальні проблеми курсу основ інформатики та їх дослідження	7
I.2 Місце інформаційно-комп'ютерних технологій у викладанні інформатики	10
I.3 Електронні конспекти уроків	12
I.4 Приклад ЕКУ	15
I.5 Електронні сценарії уроків	20
I.6 Приклад електронного сценарію уроку	23
РОЗДІЛ II	
II.1 Застосування методу проектів у процесі викладання основ інформатики	25
II.2 Вивчення алгоритмізації та програмування у формі експрес-курсу	27
РОЗДІЛ III	
III.1 Проблеми вивчення теми “Комп'ютерне моделювання. Основи алгоритмізації та програмування” у школі	29
III.2 Методичне обґрунтування експрес-курсу алгоритмізації та програмування	30
III.3 Висновки щодо принципів побудови ЕКАП	37
РОЗДІЛ IV	
IV.1 Міжпредметні зв'язки та математична складова курсу основ інформатики. Значення міжпредметних зв'язків для інформатики	41
IV.2 Бінарні уроки	45
IV.3 Зразки бінарних уроків	47
IV.4 Зв'язки інформатики з виховною роботою	54
РОЗДІЛ V	
V.1 Пошуки шляхів подолання проблем в досягненні високого рівня якості навчання	57
V.2 Обґрунтування методу аналізу якості навчання	65
V.3 Методичні поради щодо використання методу застосування коефіцієнта якості навчання у навчально-виховному процесі	66
V.4 Висновки щодо застосування технології методу аналізу якості навчання	74
РОЗДІЛ VI	
VI.1 Методична робота учителя інформатики	75
VI.2 Робота з дидактичними матеріалами як складова методичної роботи	79
VI.3 Особливості викладання основ інформатики в 9-у класі	82
РОЗДІЛ VII	
VII.1 Організація позакласної роботи з основ інформатики	93
VII.2 Про сучасні олімпіади з інформатики	95
VII.3 Якою повинна бути олімпіадна задача? Письмовий опис алгоритма та етапи розв'язування	99
VII.4 Замість післямови	103
Додатки	
1 Планування роботи ОШППД протягом 2007-2010 р.р.	105
2 Практикум з програмування	108
3 Про мотивацію навчальної діяльності	120
4 Впровадження бінарних уроків у викладання інформатики.	127
5 Викладання інформатики в 5-6 класах	149

ВСТУП

Позаду залишились три десятиріччя вивчення інформатики в загальноосвітніх школах. Цей курс завжди дуже стрімко й динамічно оновлювався, що спричинювалось швидкими змінами поколінь обчислювальної техніки та наростаючими вимогами життя в епоху переходу до інформаційної ери. Кілька років тому початок вивчення інформатики в загальноосвітніх школах було перенесено з десятого в дев'ятий клас, введено три рівні вивчення: рівень стандарту, академічний та профільний. В умовах зміни орієнтації курсу шкільної інформатики від програмування до інформаційно-комунікаційних технологій та переходу від знаннєво-змістовної до компетентнісної парадигми освіти чіткішими й актуальнішими стали проблеми вивчення курсу основ інформатики у школі. Тому протягом 2007-2010 навчальних років на базі Щасливського НВК Бориспільського району відповідно до Наказу Управління освіти і науки Київської обласної державної адміністрації від 02.04.2004 р. № 221 працювала обласна школа передового педагогічного досвіду¹ «Шляхи розв'язання актуальних проблем шкільної інформатики». Її робота була спрямована на пошуки ефективних форм і методів подолання об'єктивних проблем, які накопичились протягом двох з половиною десятиріч існування шкільного предмету «Основи інформатики». Вони виникали через недосконалість навчальних програм і підручників, недостатню кількість передбачених програмами навчальних годин, бідність та несвоєчасність методичного супроводу, а також, як уже зазначалось, завдяки небаченому розвитку комп'ютерної техніки і програмного забезпечення, що вимагало докорінних змін в інформаційній культурі та інформативній компетентності всього суспільства. Сьогодні інформатика вже вивчається паралельно із 2-го та 5-го класів, проте питання, підняті на заняттях ОШППД в 2007-2019 р.р. не перестали бути актуальними й нині, тому в цьому методичному посібнику часто будуть згадуватись і використовуватись результати роботи ОШППД.

Сучасні пошуки оптимальних шляхів організації навчально-виховного процесу сконцентровані навколо пріоритетних нині інноваційних технологій навчання (ІТН) та інформаційно-комунікаційних технологій навчання (ІКТ). З одного боку – це правильно, бо продиктовано доцільністю, адже світовий розвиток, чим далі - тим сильніше, зазнає впливу процесів глобалізації, і освітня галузь, не зважаючи на її традиційно консервативний характер, не може стояти осторонь від них. Але, з іншого боку, ці пріоритети вже по своїй природі *суб'єктивні*, тобто такі, що не можуть бути абсолютними й довготривалими. Досить згадати, для прикладу, проблеми глобального потепління в природі. Адже вони скоріше гіпотетичні, ніж наукові, бо успішно пояснюються та підтверджуються циклічністю природних процесів, які спостерігались протягом всієї історії людства.

Хоча для всіх понять намагаються дати чіткі тлумачення, абсолютних серед них не існує. Наведемо, так би мовити, канонічний приклад, про основні (неозначувані) поняття геометрії: точку, праму чи площину. Через описо-

¹ Далі вживається скорочена назва ОШППД.

вий характер різні люди, хоча б у дрібних деталях, уявляють ці поняття по різному, причому така обставина не заважає їм розуміти, що мається на увазі одне і те ж. Ще в більшій мірі це стосується похідних понять, які формулюються через основні. Отже, можна стверджувати, що й поняття «інноваційні технології навчання» та «інтерактивні методи» мають в деякій мірі приблизний зміст, а інколи їх свідомо чи не свідомо плутають. І все ж, беручи до уваги думку Р. Декарта: «Дайте точні визначення поняттям і ви позбавите людство від половини його помилок», у пошуках істини потрібно постійно намагатись уточнювати терміни. Слово *інновація* в перекладі з латинської мови означає оновлення, зміну, введення нового. У педагогічній інтерпретації *інновація* означає нововведення, що поліпшує хід і результати навчально-виховного процесу.

Оптимізм в оцінці ролі впровадження ІТН та ІКТ у навчальний процес не слід перебільшувати до ейфоричного самообману. Відігравши роль засобу *перегрупування освітніх ресурсів на базі сучасних технічних досягнень*, вони з часом стануть звичними складовими навчально-виховного процесу. Отже, інноваційні та комп'ютерні технології навчання, незважаючи на їх всеохоплюючий характер, не можуть претендувати на роль провідних напрямків розвитку освітньої галузі.

На таких міркуваннях ґрунтувався вибір проблем та шляхів їх вирішення для трирічного періоду діяльності ОШПД. По-перше, ІКТ на уроках інформатики через специфіку предмета вже давно стали звичними й ординарними засобами навчання. По-друге, використання ІКТ може бути ефективним лише при умові доцільності. По-третє, застосування інформаційно-комп'ютерних технологій є окремою проблемою навчально-виховного процесу в цілому, тому дослідження в цій галузі повинні бути винесеними за межі методики викладання основ інформатики.

Сьогодні дії надто часто підміняються *деклараціями*. Це стосується й використання інноваційних технологій навчання, адже практично всі так звані інноваційні методи навчання були давно відомими, правда, на них з часом менше акцентувалась увага. У висновках авторитетних комісій, які років 40 тому вивчали унікальний досвід знаменитого учителя математики В.Ф. Шаталова, констатувалось, що всі понад 200 дидактичних прийомів, які він широко застосовував у своїй практиці, були широко відомими. У зв'язку з цим його досвід, не зважаючи на високу результативність, тривалий час не визнавався новаторським. Але сьогодні більшість методів та прийомів із арсеналу В.Ф. Шаталова належать до інноваційних технологій навчання. Заради справедливості слід зауважити, що в ті роки носіями передового педагогічного досвіду були цілі плеяди учителів-новаторів. Варто також сказати, що тотальне, здебільшого декларативне, вживання терміну «інновації» замість впровадження досвіду учителів-новаторів у наш час приносить більше шкоди, ніж практичної користі.

Застосування вище згаданих ІКТ, нерідко також підноситься до рангу інноваційних технологій. При цьому, через недостатню підготовку широкого кола педагогічних працівників, використання комп'ютерів здебільшого зво-

диться до ролі звичайних технічних засобів навчання, які використовуються для аудіовізуального супроводу навчальних занять чи виховних заходів. Але використання лише демонстраційних функцій сучасних комп'ютерів (*демонстраційна складова*) не може виправдати навіть матеріальних затрат на них.

Ще в 60-80-і роки минулого сторіччя, коли в навчально-виховному процесі масово використовувались такі традиційні ТЗН, як кінопроектори, діаскопи, кодоскопи, програвачі, телевізори, було централізовано створено великі навчально-пізнавальні фільмотеки, фонотеки, діатеки тощо. Але дуже швидко стало зрозуміло, що все це малоефективне. Дійсно, навчальні кінофільми чи діафільми могли використовуватись лише епізодично, фрагментарно, на вступних чи підсумкових етапах вивчення теми, вимагали значного часу для монтажу і експлуатації навчальної техніки, мало відповідали конкретним умовам на уроці. Тому на практиці технічні засоби навчання застосовувались рідко. Природно, що при розгляді комп'ютера в якості демонструючого ТЗН, сучасні, навіть найкращі, ліцензовані програмно-навчальні засоби (ПНЗ) успадкували три дуже суттєві хиби тих фільмо-, фоно- та діатек. Перша – *консервативність*, адже їх важко адаптувати до потреб учителя з конкретними педагогічними ситуаціями, навпаки, використання ПНЗ змушує учителя самому пристосовуватись до них. Це *нівелює творчий характер навчального процесу*, якщо не ставити за мету провести показовий, так званий «відкритий» урок. Друга – *переважно інформаційний характер ПНЗ*. Учителю ж необхідні динамічні, із експериментальними, а не тільки демонстраційними та тестуючими можливостями програмні засоби.



схема I.1

Нарешті, третя - *всеоб'ємність*, тобто поверхове охоплення всього курсу.

Справжньою метою застосування ІКТ повинна бути *інтенсифікація розумової діяльності під час навчально-виховного процесу*. Тут маються на увазі можливості комп'ютерної техніки в моделюванні та проведенні чисельних експериментів, статистичних та аналітико-синтетичних досліджень, автоматизації пошукової роботи, класифікації й експертних оцінок, тестування і т.д., тобто *аналітичної складової* можливостей сучасної комп'ютерної техніки. Один із варіантів схеми досягнення цієї мети такий (схема І.1):

Зі схеми І.1 видно, що застосування ІКТ в навчально-виховному процесі доцільно розділяти на три етапи, які можна умовно прив'язати до вікових категорій учнів, зокрема: І етап – школа І ступеня, ІІ етап – школа ІІ ступеня, ІІІ етап – школа ІІІ ступеня. Але, залежно від категорії навчального закладу, термінів початку та рівня вивчення основ інформатики, етапи можна зміщувати.

РОЗДІЛ І

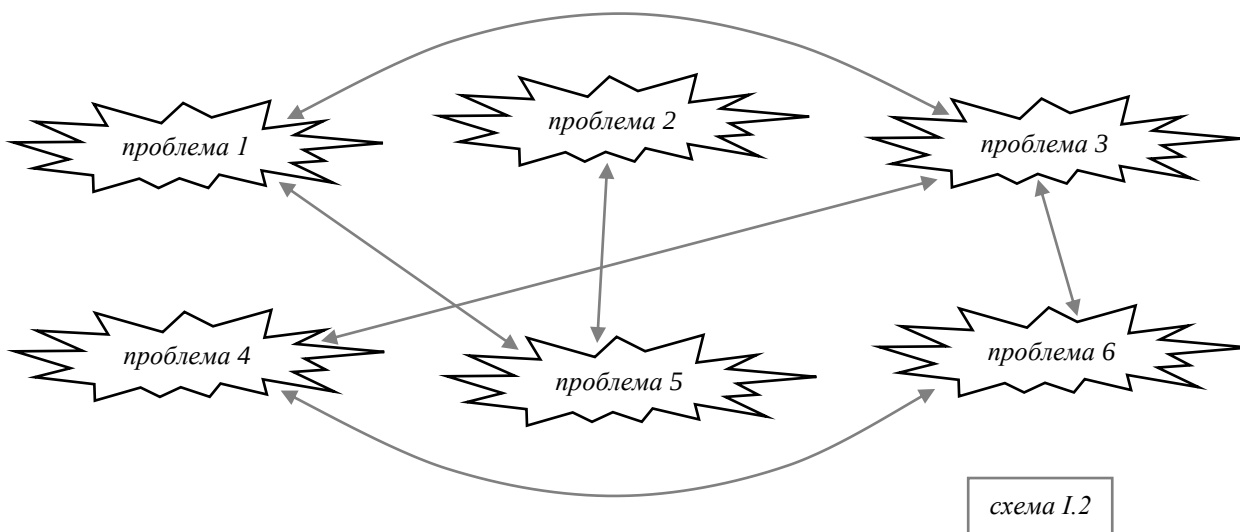
І.1 АКТУАЛЬНІ ПРОБЛЕМИ КУРСУ ОСНОВ ІНФОРМАТИКИ ТА ЇХ ДОСЛІДЖЕННЯ

З точки зору учителя-практика основними об'єктивними й актуальними проблемами сучасного курсу шкільної інформатики є (див. схему І.2):

1. *Існування встановлених діючими програмами не обґрунтованих пропорцій і акцентів між двома складовими – розділом «Курс інформаційно-комунікаційних технологій» (ІКТ - курс користувача комп'ютера) і розділом «Алгоритмізація та програмування» (АП);*
2. *Відсутність логічної й гармонійної єдності між складовими єдиного курсу – розділами інформаційно-комунікаційних технологій та моделюванням і основами алгоритмізації й програмування;*
3. *Відсутність необхідного рівня міжпредметних зв'язків курсу «Основи інформатики» з іншими навчальними дисциплінами, у першу чергу з точними та природничими;*
4. *Відсутність правильного і чіткого усвідомлення місця курсу основ інформатики в системі загальної середньої освіти та у формуванні гармонійно розвиненої особистості;*
5. *Відсутність спільних теоретичних та практичних основ між курсом інформатики і позакласною роботою з цього предмету.*

У 2009-2010 н.р. в зв'язку із введенням предмета «Основи інформатики» в навчальні плани для 9-х класів виникла ще одна проблема:

6. *Відсутність якісної навчальної літератури, методичного і дидактичного забезпечення викладання основ інформатики в умовах 12-річної школи.*



Остання проблема ще більше загострюється у зв'язку із заміною 12-річного терміну навчання на одинадцятирічний, початком вивчення із 2012 року курсу «Сходинки до інформатики» для 2-4 класів та, в перспективі, основ інформатики у 5-9 класах. В таких умовах зовсім неясно, якою буде шкільна інформатика в 10-11 класах, коли і які з'являться підручники, навчально-методичні і дидактичні посібники.

Окремо стоїть ще одна надзвичайно важлива проблема:

7. *Наявність у впливових колах МОН молоді спорту та Академії педагогічних наук намірів зміни освітньої парадигми від знань-во-змістовної до компетентнісної, які безперечно негативно вплинуть не тільки на рівень знань з основ інформатики, а й на формування інформаційної культури в цілому.*

Перші шість щойно перерахованих проблем взаємопов'язані (див. схему I.2). Дійсно, необхідна зміна пропорції між розділами «Практичний курс інформатики (курс користувача комп'ютера)» і «Алгоритмізація та програмування» в сторону збільшення долі останнього (проблема 1), але це приведе до збільшення присутності математичної компоненти в шкільному курсі основ інформатики, адже алгоритмізація та програмування – це, по суті, прикладна математика (проблема 3). Сьогодні ж курс інформатики віднесений до категорії предметів «Технології». Вкрай необхідне встановлення тіснішого зв'язку між вищеназваними складовими шкільного курсу інформатики (проблема 3), у свою чергу це приведе до фактичного проникнення програмування, у сучасному варіанті візуального проектування та об'єктно-орієнтованого програмування, у більшість тем курсу основ інформатики (проблема 1). Особливо актуальна сьогодні потреба формування сучасної інформаційної культури та інформативної компетентності учнів (проблема 4) залежить від усвідомлення місця курсу основ інформатики в навчально-виховному процесі (проблема 3). Нарешті, без ефективної позакласної роботи з предмету (проблема 5), яка дуже залежить від рівня культури програмування (проблема 1) та взаємопов'язаності вміння програмувати з навичками користувача (проблема 2) не можливо якісно організувати вивчення основ інформатики в ці-

лому. Очевидно, що проблеми 3 та 4 мають прямий зв'язок з проблемою 6. Виходячи з цього, сукупність проблем 1 - 6 доцільно сприймати, як *єдину і масштабну проблему*.

Тому завдання, яке ставила перед собою ОШППД, мусило полягати в тому, щоб, спираючись на досвід керівника школи та її слухачів, забезпечити ефективні пошуки шляхів розв'язання цієї масштабної проблеми.

Як і будь-який вузький напрям дослідження в галузі методики, сформульовані проблеми не можна розглядати не в контексті вивчення якості навчання та мотивації навчальної діяльності учнів. Тому нижче цим питанням буде також приділено увагу.

Необхідність пошуків ефективних форм і методів подолання названих проблем визначає основні напрямки роботи ОШППД:

1. Удосконалення методики проведення урочної та позакласної роботи з основ інформатики *в умовах інтенсивного використання інформаційно-комп'ютерних та інтерактивних технологій навчання*;
2. Розробка сучасних електронних дидактичних засобів навчання з основ інформатики у середовищі *Microsoft Office*;
3. Вивчення шляхів удосконалення організації позакласної роботи з основ інформатики;
4. Розробка методики поєднання індивідуальних та інтерактивних методів навчання у процесі викладання інформатики.

Отже, ІТН та ІКТ належить до основних напрямків роботи ОШППД (виділено курсивом у напрямку 1), тому вони *є не предметом, а засобами навчальної та дослідницької діяльності*.

Діяльність ОШППД ґрунтується на багаторічному досвіді роботи, провідною ідеєю якого є *оптимізація викладання курсу основ інформатики з метою подолання актуальних проблем в умовах впровадження інформаційно-комп'ютерних технологій навчання, формування сучасної інформаційної культури та інформативної компетентності підростаючого покоління*. Актуальність досвіду впливає з *потреби піднесення ролі курсу основ інформатики для формування сучасної інформаційної культури та інформативної компетентності учнів*.

ОШППД ставила за мету досягнення якісних результатів:

- а) У підвищенні рівня професійної майстерності вчителів інформатики;
- б) У формуванні сучасної інформаційної культури та інформативної компетентності учнів;
- в) У підвищенні рівня культури програмування вчителів та учнів.

Протягом трьох навчальних років у процесі роботи ОШППД було досліджено теми²:

² План проведення занять ОШППД міститься в додатку 1.

- I. Сучасні проблеми у викладанні основ інформатики;
- II. Місце ІКТ у викладанні основ інформатики;
- III. Застосування методу проектів у процесі викладання основ інформатики;
- IV. Вивчення теми «Алгоритмізація та програмування» у формі експрес-курсу;
- V. Міжпредметні зв'язки та математична складова курсу основ інформатики;
- VI. Методична робота учителя інформатики;
- VII. Особливості викладання основ інформатики в 9-у класі за новими програмою та підручниками;
- VIII. Методичні особливості вивчення комп'ютерних мереж в 9 класі;
- IX. Методичні особливості вивчення векторної графіки в 9 класі;
- X. Організація позакласної роботи з основ інформатики.

Розгляд першої теми передбачав ініціювання основних актуальних проблем 1-5 курсу інформатики та шляхів їх розв'язування, тобто постановку загальних завдань перед ОШПІД ([1], [2], [3])³.

Зміст другої теми полягав у встановленні ролі ІКТ та ІТН, як засобів навчально-виховної роботи з основ інформатики. Акцентувалось на навчально-дидактичних можливостях сучасних комп'ютерів, співвідношенні між їх демонстраційною та аналітичною складовими. Розглянуто питання ефективності ліцензованих програмних засобів навчального призначення (ППЗ НП) з огляду на низькі ступені їх універсальності та своєчасності, а також складності використання, пов'язані з отриманням ліцензійних копій. Зроблено висновок 1: найбільш ефективними програмними засобами при вивченні основ інформатики на сучасному рівні ІКТ можуть стати так звані *електронні дидактичні засоби навчання* (ЕДЗ)⁴ та їх окремий різновид – *комплексні електронні дидактичні засоби* (КЕДЗ) ([4]). До складу ЕДЗ можна включити: електронні сценарії уроків (ЕСУ), електронні конспекти уроків (ЕКУ), електронні роздавальні матеріали (ЕРМ).

1.2 МІСЦЕ ІНФОРМАЦІЙНО-КОМП'ЮТЕРНИХ ТЕХНОЛОГІЙ У ВИКЛАДАННІ ІНФОРМАТИКИ

Спочатку зауважимо, що тут розглядаються лише саморобні ЕДЗ, про які мовилось вище, адже методичні поради по застосуванню ліцензованих ППЗ НП повинні включатись до їх складу, як частина, і захищатись авторським правом.

³ У квадратних дужках вказується номер використаного джерела згідно списку літератури.

⁴ Поняття *електронних дидактичних засобів навчання* та їх окремих видів, а також умовні позначення введено автором для зручності опису.

На уроках основ інформатики діючими програмами рекомендується використовувати практичну форму занять, тобто, на *кожному* уроці *кожен* учень повинен якомога більше працювати *безпосередньо* з комп'ютером. Якщо не відходити від цього принципу, слід погодитись, що застосування проекторів, інтерактивних дошок та демонстраційних екранів на уроках інформатики стає зайвим. Тим більше, що є чудова можливість користуватись локальною мережею, з якою учні досить детально знайомляться ще на початкових етапах вивчення основ інформатики. Серед системного програмного забезпечення є ряд спеціальних програм, що дають можливість організувати дистанційне керування робочими місцями учнів (РМУ) з центрального комп'ютера локальної мережі. Як оптимальний варіант можна запропонувати *Net OpSchool* (бажано нові версії). Перш за все з його допомогою учитель зможе легко організувати демонстрацію потрібної порції навчального матеріалу, що у поєднанні мовного супроводу учителя із маніпуляціями курсором миші і вказівниками на екрані дає більший ефект, ніж використання демонстраційного екрану. Адже така форма роботи спонукає учнів привчатись працювати індивідуально і разом з тим постійно концентрувати свою увагу на діях учителя, а це більш продуктивно, ніж коли вони у складі групи спостерігають за спільним екраном. Програма *Net OpSchool* дає ще багато дидактичних можливостей, у тому числі спостерігати за подіями на екранах РМУ, блокувати лише вибрані з них, застосовувати режим індивідуального управління кожним комп'ютером, робити аудіо та візуальні акценти, проводити тестування та спілкування за зразком чату тощо.

Отже під час уроку з використанням *Net OpSchool* види колективної роботи легко поєднувати з індивідуальною практикою учнів. Суттєвим недоліком ліцензованих ППЗ НП є неможливість ефективного застосування *Net OpSchool*, адже вони підпорядковують собі ресурси локальної мережі. Задум описаних нижче видів ЕДЗ, навпаки, передбачає його постійне застосування. Враховуючи можливості самостійного виготовлення і редагування а також роботу у звичному для пересічних користувачів (якими є учителі й учні) середовищі *Microsoft Office*, ЕДЗ стають не тільки конкурентноздатною альтернативою застосуванню ліцензованих ППЗ НП, а *інноваційною формою використання ІКТ* в навчально-виховному процесі.

ЕДЗ, під якими слід розуміти файли чи пакети файлів, що містять будь-яку інформацію навчального характеру – це дидактичні або наочні засоби, що розсилаються на робочі місця учнів (РМУ) через локальну мережу для використання учнями під час роботи з комп'ютером. ЕДЗ, як уже зазначалось, не слід плутати із ліцензованими програмними продуктами навчального призначення, бо вони виготовляється учителем, гнучкі за характером будови та застосування, можуть оперативно змінюватись, включати гіперпосилання, анімаційні чи звукові ефекти, мобільно адаптуватись до конкретної ситуації, мають, як правило, локальний зміст і характеризуються багатофункціональністю.

ЕДЗ можуть містити теоретичні відомості з необхідними питаннями, завданнями, тестами, бути оснащеними таблицями, схемами, малюнками.

Завдяки системі гіперпосилань ці документи мають динамічний характер, розсилка на РМУ через локальну мережу дає змогу використовувати їх оперативно і комплексно з системами програмування та навчальними пакетами. Бажано, щоб кожен ЕДЗ охоплював вузьку тему, тоді створювати його легко, при потребі завжди можна доопрацювати і удосконалити. Крім того, робота з ним урізноманітнить урок, зробить його більш динамічним, створить атмосферу позитивної мотивації вивчення навчального матеріалу, дасть можливість відчувати курс інформатики, як єдине ціле, учні матимуть можливість майже безперервно працювати з комп'ютером, не відчуючи нав'язливої ролі учителя. Такі форми роботи на уроці спонукають і вчителя підходити до більшості тем не формально, строго відповідно програмі, а із спрямуванням на наступне практичне застосування, все щойно вивчене при першій нагоді використовувати практично.

Постійно використовуючи ІКТ, від електронних дидактичних засобів, що носять локальний за змістом характер, можна безболісно переходити до великих за обсягом і змістом *комплексних електронних дидактичних засобів* (КЕДЗ). При творчому підході вони будуть ефективнішими за ліцензовані ПНЗ, маючи надзвичайно важливі переваги перед останніми: можливості конструювання, доповнення та удосконалення на будь-якому етапі.

Варто звернути увагу ще на одну важливу деталь. Справа в тому, що описані форми роботи на уроках інформатики реалізують безпаперову технологію навчання, коли учні не потребують ні підручника, ні зошита, а привчаються працювати лише з консоллю (екраном, клавіатурою та ручним маніпулятором). Така форма економічно доцільна і сприяє ефективнішому формуванню сучасної інформаційної культури та інформативної компетентності. Детальніше повернемося до неї нижче.

1.3 ЕЛЕКТРОННІ КОНСПЕКТИ УРОКІВ

Описані нижче *електронні конспекти уроків* (ЕКУ) можна розглядати, як основну форму ЕДЗ на уроці. Тому детальніше зупинимось на *понятті* ЕКУ. Зміст значною мірою впливає з назви, тобто, перш за все це конспект уроку. Але якщо його не ототожнювати з терміном «поурочний план», а мати на увазі достатньо деталізований опис відповідного темі навчального матеріалу з прикладами та методичними порадами (ремарками) щодо вивчення цього матеріалу, то такі конспекти уроків можна успішно використовувати і як фрагменти підручника для учнів, особливо в умовах недостатнього забезпечення учнів якісними підручниками, і як керівництво до дії для вчителя. Необхідно лише «сховати» від учнів методичні ремарки. Кожен виготовлений засобами *Microsoft Office* друкований конспект уроку, подібно ЕСУ, повинен містити основну частину - ядро⁵. ЕКУ з допомогою локальної мережі слід розсилати на РМУ. Використання таких конспектів буде вирішувати одночасно кілька дидактичних завдань. По-перше, учні матимуть

⁵ На відміну від ЕСУ, ядром яких є навчальні презентації, ядро ЕКУ доцільно створювати у формі документа *Microsoft Word*, який найзручніше піддається прокрутці.

можливість читати, так би мовити, в оригіналі думки учителя, причому навіть під час підготовки домашнього завдання, якщо скопіювати конспект на ГМД. По-друге, працюючи над темою, кожен учень буде постійно комплексно застосовувати знання, уміння та практичні навички, одержані на уроках інформатики раніше, що буде для них додатковим чинником у формуванні сучасної інформаційної культури.

В електронних конспектах уроків широко застосовуються як графічні, так і анімаційні засоби та гіперпосилання. Це дає можливість роботу над навчальним текстом із статичного стану перевести в динамічний, що має дуже великі, раніше недоступні, дидактичні можливості. Для створення ЕКУ можна використовувати MS Word чи MS Power Point, а з допомогою кнопок переходів та гіперпосилань приєднувати до ядра інші текстові та графічні файли і навіть цілком автономні додатки.

Як уже згадувалось вище, на прикладі електронного конспекту уроку вводиться якісно нове поняття *комплексного електронного дидактичного засобу*.

Для оцінювання якості ЕСУ та ЕКУ можна користуватись умовними критеріями. Спочатку наведемо зразок для ЕКУ⁶: *таблиця II.1*

<i>№</i>	<i>критерій</i>	<i>оцінка в балах</i>
1	Ступінь охоплення навчального матеріалу	__ балів
2	Логічність побудови та якість оформлення	__ балів
3	Відповідність загальним вимогам до конспекту уроку, лекції	__ балів
4	Відповідність загальним вимогам до електронних навчальних засобів	__ балів
5	Відокремлення навчальних матеріалів від методичних рекомендацій	__ балів
6	Ступінь структурованості	__ балів
6.1	Якість ядра	__ балів
6.2	Наявність додатків	__ балів
7	Навігаційні можливості, ознаки гіпертекстового документа	__ балів
7.1	Гіперпосилання на закладки	__ балів
7.2	Гіперпосилання на файли	__ балів
7.3	Гіперпосилання на інтернетресурси	__ балів
8	Можливості редагування	__ балів
9	Інтеграційні можливості з іншими ЕДЗ	__ балів
10	Можливості закріплення та тестування	__ балів
10.1	Наявність та наочність прикладів	__ балів
10.2	Наявність і форма запитань для закріплення	__ балів

⁶ Кожен пункт та підпункти оцінюються по десятибальній шкалі. Загальна оцінка виставляється по десятибальній шкалі (10% балів від загального числа – 1 бал).

10. 3	Наявність завдань для закріплення (робочий бланк відповідей)	___ балів
	Загальна кількість балів	___ балів

Коротко прокоментуємо запропонований зразок критеріїв. Перш за все зауважимо, що вони необхідні тому, хто створює та використовує ЕКУ, адже і в процесі створення і для використання необхідно мати інструменти оцінювання. Більшість пунктів очевидні, тому прокоментуємо тільки деякі нюанси оцінювання, які стосуються лише ЕКУ.

Перший пункт дає оцінку об'єму навчального матеріалу, представленому в ЕКУ. Він, як правило, повинен відповідати одному уроку, одній темі чи одному розділу. Але залежно від обсягу матеріалу, ЕКУ повинен мати чітке призначення: повторення, вивчення нового матеріалу, закріплення, узагальнення чи перевірки знань, наприклад, коли ЕКУ охоплює навчальний матеріал цілого розділу, то його тип може бути: урок повторення, урок узагальнення або урок перевірки знань. Як видно із наведеного зразка критеріїв, п.1 прямо пов'язаний з п. 3; 5; 6, тому виставлення балів у цих пунктах залежить від ступеня охоплення навчального матеріалу і оцінки п.1. Отже, необхідно, щоб ЕКУ охоплював максимально ефективний за обсягом навчальний матеріал.

Виходячи з того, що тут ідеться про план навчального заняття, пункти 2 та 3 необхідні лише для контролю відповідності ЕКУ загальним критеріям оцінювання уроків. Пункт 4 призначений для контролю відповідності загально-прийнятим порадам щодо розміщення матеріалу в електронних документах (презентації, публікації, текстові документи, web-сторінки тощо), тобто зовнішнього вигляду сторінки чи кадру, палітри кольорів, розміру та гарнітури шрифту, кількості тексту і т.д.

З допомогою п. 5 контролюється чи відокремлені навчальні матеріали, призначені для учнів від специфічних, але не потрібних учням методичних порад та коментарів, які адресовані учителям та лекторам.

У пункті 6 мається на увазі кількість видів структурних компонентів ядра та додаткових електронних джерел. Може бути однорівнева (проста) структура, вона передбачає наявність пунктів плану уроку та підпунктів до них. Дворівневе або кількарівневе структурування передбачає наявність посилань на додатки в кінці документа або в інших файлах.

Деякі пункти, наприклад п. 7, мають свою структуру, бо є необхідність не тільки загальної оцінки, а й контролю за наявністю елементів. Навігаційні можливості – це послуги для користувачів, що дають змогу вільно пересуватись по ЕКУ, а основними засобами навігації в документах *Microsoft Word* є гіперпосилання. Слід зауважити, що ЕКУ, як і більшість інших типів ЕДЗ, немислимі без гіперпосилань та засобів навігації, причому для ЕКУ вони особливо важливі, адже тільки ядро може містити достатньо багато сторінок.

Залежно від мети конкретного ЕКУ можуть бути необхідними можливості санкціонованого (захищеного) доступу до редагування та узгодження для спільного використання з іншими ЕДЗ. Це передбачено п. 8-9.

Однією з найсуттєвіших ознак кожного ЕКУ є засоби закріплення та тестування, тому в критерії включено п.10 з окремими підпунктами. Приклади та запитання для закріплення в електронних документах можуть бути більш наочними та ефективними за рахунок включення елементів комп'ютерної графіки (п. 10.1-10.2). Під робочими бланками відповідей (п.10.3) маються на увазі документи, що містять індивідуальні завдання з розподілом на варіанти до конкретно взятого ЕКУ, які можуть мати вигляд окремої сторінки робочого електронного зошита. При створенні серії ЕКУ до окремої теми можна у вигляді додатку підготувати електронний робочий зошит, який можна буде фрагментарно використовувати протягом вивчення цілої теми, якій відповідає серія ЕКУ.

Кожен учитель може поступово створити цілу бібліотеку ЕКУ, об'єднавши які та відповідні їм електронні робочі зошити з самостійними, практичними та контрольними роботами, одержати цілий електронний курс з окремо вибраної теми, який можна класифікувати як комплексний електронний дидактичний засіб. У якості зразка КЕДЗ під час розгляду другої теми ОШПД (друге заняття) розглядався приклад дидактичної довідкової системи «Турбо Паскаль у таблицях».

1.4 ПРИКЛАД ЕКУ

(окремий файл № 1 - ядро) **ЛІНІЙНІ ПРОГРАМИ**

Тема, мета, обладнання, література ❶1

ПЛАН УРОКУ

I. Повторення та перевірка домашнього завдання

Дати відповіді на питання:

- 1) Що розуміють під поняттям структури?
- 2) Користуючись додатком 2 пояснити, яку структуру має програма на мовою Паскаль.
- 3) Що розуміють під оголошенням та описом величин?
- 4) Користуючись додатком 3 пояснити різні способи оголошення та опису величин.

II. Теоретичні відомості

Лінійними називають алгоритми та програми, якщо їх команди виконуються у порядку запису, тобто:

команда 1;	виконується першою
команда 2;	виконується другою
команда 3;	виконується третьою
...	
команда n-1;	виконується передостанньою
команда n;	виконується останньою

Звідси зрозуміло, що лінійні алгоритми та програми можуть містити лише команди вводу даних, присвоєння, виклику підпрограми ❷2 та

виводу результатів⁷ (див. зноску), тому для розгляду і написання лінійних алгоритмів і програм, крім знання структури **У3** алгоритму та програми, правил оголошення та опису величин, **У4** необхідно знати формат та правила виконання команд вводу **У5**, виводу **У6** та присвоєння **У7**.

III. Приклад

```

program LineProgr;    {заголовок програми}
  uses Crt;            {блок опису модулів}
  var a,b,p,s:real;    {блок оголошення і опису величин}
begin                 {початок тіла програми}
  WriteLn('a?');ReadLn(a);    {блок вводу даних}
  WriteLn('b?');ReadLn(b);
  p:=2*(a+b);           {блок опрацювання інформації}
  s:=a*b;
  WriteLn('p=',p:6:2,'; s=',s:6:2); {блок виводу результатів}
end.                   {кінець тіла програми}

```

IV. Практична робота

- 1) Користуючись гіперпосиланням
{запуск програми з допомогою транслятора PasABC **У8}**
- 2) Створити документ D:\klass\11_a\Group_1\Практична робота.doc згідно додатку 8, виконавши запропоновані завдання. **У9**

завантажити та виконати зразок лінійної програми в середовищі PasABC покроково (F8) та прослідкувати послідовність виконання команд. Переконайтесь, що ця програма – лінійна.

V. Домашнє завдання

Повторити: Структуру програми мовою Паскаль;
Оголошення та опис величин у програмах мовою Паскаль.

Вивчити: Поняття лінійної програми;
Формат і особливості виконання команд вводу/виводу;
Формат і правило виконання команди присвоєння;

Практичне завдання : Написати програму визначення довжини кола і площі круга за введеним з клавіатури значенням радіуса кола.

VI. Додатки:

(окремий файл № 2)

додаток 1 **У1**

Тема

- Лінійна програма та її структура;
- Способи вводу даних та виводу результатів, вікно Crt;
- Коментарі у програмі;

⁷ Команди вводу та виводу розглядаються, як прості команди, але кожна з них по суті є допоміжною програмою (підпрограмою). Підпрограми поділяються на стандартні та створені користувачем. Запис команд вводу/виводу у програмі насправді є командами виклику відповідних підпрограм. Тому команду виклику **У2** будь-якої підпрограми слід розглядати як просту команду.

- Покрокове виконання програми;

Мета:

- Познайти учнів із прикладами лінійних програм, сформувати навички складання і запису лінійних програм, вставки коментарів у програмах, навички виконання та покрокового виконання програм;
- На прикладі модуля *Crt* познайти учнів із поняттям модуля.

Обладнання:

- Інтерпретатор мови програмування Паскаль *PascalABC*⁸;
- Програма для управління локальною мережею *NetOp School*, версія 5.0⁹;
- Таблиці для демонстрації структури програми, оголошення та опису величин і прикладів лінійної програми;

Література: І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов, Інформатика, частина 2, Київ, “Форум”, 2004р.

(окремий файл 3)

додаток 2

СТРУКТУРА ПРОГРАМИ НА PasABC 03

```
[program ім'я;]
[uses імена модулів{через кому};]
[const ім'я1 = значення; ім'я2 = значення; . .]
[type опис нестандартних типів]
[var ім'я1 : тип; ім'я1 : тип;] {однотипні – через кому}
[блок опису підпрограм]
begin
  [тіло програми]
end.
```

блок
оголошень
та описів

текст, взятий
в квадратні дужки,
може бути відсутній

(окремий файл 4)

додаток 3

ОГОЛОШЕННЯ І ОПИС ВЕЛИЧИН 04

```
const n=5;
      M : array[1..n] of 0..9 = (8,2,1,0,7);
```

ТИП ЕЛЕМЕНТІВ МАСИВА - ДІАПАЗОН

```
type Mas = array[1..n,1..n] of real;
var a, b, c, max : real;
```

ТАК ОГОЛОШУЮТЬСЯ КІЛЬКА
ОДНОТИПНИХ ВЕЛИЧИН

```
st : string; d : byte;
Mas1 : array[1..n,1..n] of integer;
Mas2 : Mas;
```

ВЕЛИЧИНА Mas2
МАС ОГОЛОШЕНИЙ ТИП
array[1..n,1..n] of real;

(окремий файл № 5)

додаток

4

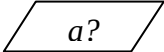
⁸ При вивченні теми «Алгоритмізація і програмування» у базовому курсі основ інформатики використання одного із сучасних трансляторів мови програмування, створених під операційну систему *Windows*, наприклад *PascalABC*, значно ефективніше за використання сьогодні застарілих *Turbo Pascal* чи *Borland Pascal*.

⁹ На наше переконання сучасний урок основ інформатики в умовах практичного проведення кожного заняття.

команда вводу 5:

Команда вводу в мові програмування Паскаль називається процедурою вводу.

Словесно блок схемою на алгоритмічній мові на мові паскаль

Ввести *a*  ввести(*a*) `Read(a);`
`Read(a,b,...);`
`ReadLn(a);`
`ReadLn(a,b,...);`

додаток 5

команда присвоєння 7:

Формат

правило виконання

Ім'я величини := вираз

Вираз записується за правилами лінійного запису, може містити змінні величини.

Команда виконується так:

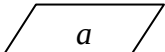
- величинам у виразі надаються їх значення;
- визначається значення виразу;
- одержане значення надається величині, ім'я якої записано зліва від знака " := "

додаток 6

команда виводу 6:

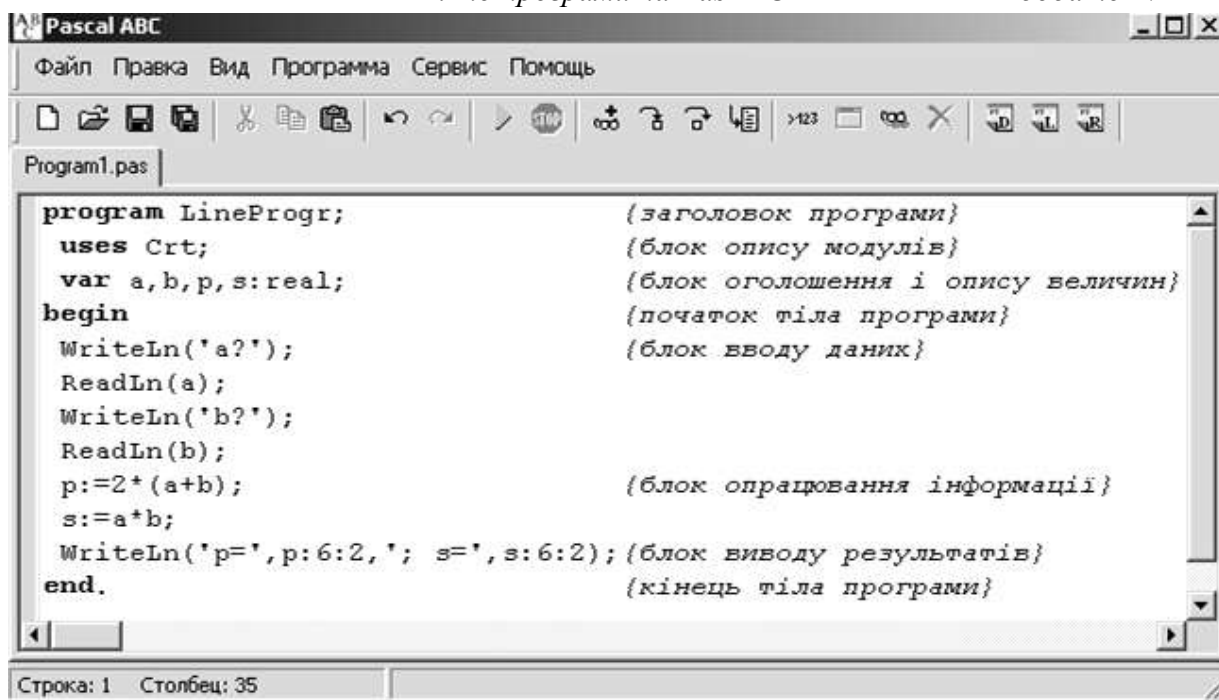
Команда виводу в мові програмування Паскаль називається процедурою виводу.

Словесно блок схемою на алгоритмічній мові на мові паскаль

вивести *a*  вивести(*a*) `Write(a);`
`Write(a,b,...);`
`WriteLn(a);`
`WriteLn(a,b,...);`

Вікно програми на PasABC

додаток 7 8



```
program LineProgr;                                {заголовок програми}
uses Crt;                                          {блок опису модулів}
var a,b,p,s:real;                                {блок оголошення і опису величин}
begin                                             {початок тіла програми}
  WriteLn('a?');                                {блок вводу даних}
  ReadLn(a);
  WriteLn('b?');
  ReadLn(b);
  p:=2*(a+b);                                    {блок опрацювання інформації}
  s:=a*b;
  WriteLn('p=',p:6:2,'; s=',s:6:2); {блок виводу результатів}
end.                                             {кінець тіла програми}
```

(окремий файл № 6)

додаток 8 9

Завдання для практичної роботи¹⁰:

№	I варіант	Бали	II варіант	бали
1	Чи може програма мати вигляд: <i>begin end.</i> так/ні (залишіть правильну відповідь)	0/1	Чи може програма мати вигляд: <i>begin end;</i> так/ні (залишіть правильну відповідь)	0/1
2	обґрунтуйте відповідь:	0/1	обґрунтуйте відповідь:	0/1
3	Чи може бути у програмі команда: <i>begin end;</i> так/ні (залишіть правильну відповідь)	0/1	Чи може бути у програмі команда: <i>begin end.</i> так/ні (залишіть правильну відповідь)	0/1
4	обґрунтуйте відповідь:	0/1	обґрунтуйте відповідь:	0/1
5	Порівняйте записи: 1) <i>var a,b:integer,p,s:real;</i> 2) <i>var a,b,p,s:real;</i> Який з них правильний? 1/2 (залишіть правильну відповідь)	0/1	Порівняйте записи: 1) <i>var a;b;p;s:real;</i> 2) <i>var a,b:integer;p,s:real;</i> Який з них правильний? 1/2 (залишіть правильну відповідь)	0/1
6	обґрунтуйте відповідь:	0/1	обґрунтуйте відповідь:	0/1
7	Порівняйте записи: <i>(1+n)*n/2:=Sn;</i> та <i>Sn:=(1+n)*n/2;</i> Який з них не є командою? 1/2 (залишіть правильну відповідь)	0/1	Порівняйте записи: <i>Sn:=(1+n)*n/2;</i> та <i>Sn:=(1+n)*n/2</i> Який з них не є командою? 1/2 (залишіть правильну відповідь)	0/1
8	обґрунтуйте відповідь:	0/1	обґрунтуйте відповідь:	0/1
9	Яка з команд правильна: <i>Read(); Write;</i> 1/2 (залишіть правильну відповідь)	0/1	Яка з команд правильна: <i>Read; Write();</i> 1/2 (залишіть правильну відповідь)	0/1
10	обґрунтуйте відповідь:	0/1	обґрунтуйте відповідь:	0/1
11	Пояснити команду: <i>WriteLn(Sn);</i>	0/1	Пояснити команду: <i>WriteLn('Sn=',Sn);</i>	0/1
12	Пояснити команду: <i>Write('Sn=',Sn);</i>	0/1	Пояснити команду: <i>Write('Sn=',Sn:12:1);</i>	0/1
	Загальна кількість балів:	0/12	Загальна кількість балів:	0/12

Форма запису електронних конспектів уроку дещо незвична у порівнянні з традиційним виглядом поурочного плану, адже тут обов'язкові гіперпосилання, ілюстрації, додатки тощо. ЕКУ – це найчастіше пакет електронних документів різного формату, призначених для використання на уроці з допомогою локальної комп'ютерної мережі, причому, ці документи мають, так би мовити, різні області видимості. У наведеному вище прикладі, як уже зазначалось, додаток 1 призначений для викладача, тому його слід зробити невидимим для учнів. Подібно до цього, ЕКУ може містити цілий ряд адресованих тільки учителеві методичних ремарок.

Виходячи з потреби описати електронний дидактичний засіб, у прикладі з допомогою значків зразка **U1**, **O1** позначено гіперпосилання, які візуально такі ж, як і в справжньому ЕКУ¹¹. Додатки, розміщені в окремих файлах, тут помічені таким чином: (окремий файл № 1). Особливо звертаємо увагу на гіперпосилання запуск програми з допомогою транслятора PasABC**U8** та відповідний йому додаток 7. Таким чином у прикладі зображено вікно системи програмування *PasABC*, що використовується для самостійного закріплення вивченого матеріалу. За реальних обставин гіперпосилання **U8** дійсно запускає на виконання інтерпретатор *PasABC*, але, нажаль, учневі самостійно необхідно відкрити програму, яку ми бачимо в додатку 7. Так само, нажаль

¹⁰ Наведені тут завдання для практичної роботи можна взяти за зразок, адже у жодному з нових підручників з інформатики [18], [19], [20] немає подібних практичних чи лабораторних робіт з 12-ма дворівневими завданнями, які поєднують тестові й логічні завдання і оцінюються рівно 12-ма балами.

¹¹ На відміну від природного для позначення гіперпосилання символа **U** позначення **U1**, **O1** вибрано через їх зрозумілий додатковий навігаційний зміст.

перехід до інтерпретатора *PasABC* не має зворотного зв'язку. Уникнути таких незручностей у принципі можна, застосувавши замість системи *PasABC*, яка є інтерпретатором, компілятор, зокрема *Delphi*. Тоді гіперпосилання 08 зверталось би до файлу формату **exe*. Але нагальної дидактичної потреби в цьому немає, тому не будемо концентруватись на тому, що зв'язок з додатком, який не входить до пакету *Microsoft Office*, однонаправлений. Нижче, при описі ЕДЗ, будемо дотримуватись подібних позначень. Для практичного засвоєння радимо згідно наведеного опису створити ЕКУ і випробувати його в режимі виконання. Пропонуємо також оцінити ЕКУ "Лінійні програми" згідно наведених у таблиці 1 критеріїв.

1.5 ЕЛЕКТРОННІ СЦЕНАРІЇ УРОКІВ

Описані нижче ЕСУ суттєво відрізняються від ЕКУ як за формою, так і за призначенням. Під **електронними сценаріями уроків** тут розуміються комп'ютерні презентації, але на відміну від традиційних презентацій - аудіо-візуальних супроводів до окремих навчальних занять чи виховних заходів, вони є *сценаріями* для проведення заняття, частково колективно, під керівництвом учителя, частково самостійно під наглядом останнього. Використання традиційної презентації – це пасивне навчання, адже учні не мають можливостей для втручання в процес демонстрації, їх роль зводиться лише до спостереження. ЕСУ – це *алгоритм дій учнів* по вивченню оголошеної теми із завданнями, запитаннями, ілюстраціями та прикладами. ЕСУ – це сценарій, тому всю заплановану на уроці роботу слід повністю відображати в ньому.

ЕСУ близький до електронного конспекту уроку, але, на відміну від нього, не містить блоків, призначених для учителя: запису мети, опису обладнання, теоретичних відомостей з методичними порадами, хронології дій учителя та учнів, які притаманні традиційним поурочним планам і їх електронним аналогам. Він може використовуватись паралельно з ЕКУ, але не обов'язково, адже включає в себе всі елементи ЕКУ, необхідні для роботи з учнями. Роль учителя на уроці з використанням ЕСУ зводиться до ролі арбітра, що займається хронометражем, оцінюванням, коментарями тощо.

Електронні сценарії уроків містять загальний план всього заняття, зміст його окремих структурних елементів, запитання та вправи для колективного і самостійного закріплення, домашні завдання, гіперпосилання на додаткові електронні джерела до даної теми. Щоб охопити тему, в *ядро* ЕСУ, створене з допомогою *Microsoft PowerPoint*, необхідно включити достатньо багато кадрів, часто із складною ієрархією. Цього можна досягти в одній презентації, де кожен структурний елемент становить один слайд, або в групі взаємопов'язаних презентацій. Але у будь-якому випадку елементи ЕСУ слід зв'язувати з допомогою кнопок переходу та гіперпосилань. Дуже корисними і ефективними можуть бути й перемикачі (тригери). Зрозуміло, що ядро ЕСУ створюється власними зусиллями учителя, можливо спільно з іншими учителями чи учнями.

Вже не один рік масово проходить навчання учителів по програмі «Інтел® Навчання для майбутнього», в основу якої покладено метод проектів. Свого часу переважно всі вчителі інформатики брали участь в тренінгах чи майстер-класах Інтел, отримуючи відповідні сертифікати. Навчання по програмі «Інтел® Навчання для майбутнього» частіше всього проходить як імітація проектної роботи вчителя з учнями, слухач готує портфоліо, в яке поміщає не тільки всі елементи, які належить готувати вчителю, а й так звані «учнівські» презентації, публікації та веб-сайти, виготовлені учителем від імені учнів, а потім ще й оцінені ним же. На цьому, як правило проектна робота «з учнями» й закінчується. Вважається, що учитель «навчився» організовувати роботу учнями за методом проектів та з використанням ІКТ. Правда на уроках застосувати цю технологію неможливо, адже жодна навчальна програма не передбачає відводити час уроків на створення разом з учнями цілісного проекту на конкретну тему. Досить анекдотично, чи не правда? Проектну роботу з учнями з використанням ІКТ можна реалізувати тільки в позаурочний час. Однією з форм такої роботи, не ігрової, а навчальної, корисної не тільки учням, а й учителям, може бути виготовлення ЕСУ.

У ядрі ЕСУ майже обов'язкові гіперпосилання на *допоміжні* ЕДЗ, створені з допомогою компонентів пакета *Microsoft Office* чи інших додатків *Windows* та автономні частини ліцензованих ППЗ НП, які можна імпортувати також як мультимедійні компоненти в окремі слайди. Проте, доцільніше їх підключати до ЕСУ з допомогою гіперпосилань, це допоможе уникнути переобтяження ядра, що часто приводить до аварійних зупинок, тобто "зависань". Пропонуємо скористатись порадою. Якщо презентацію зберегти у форматі *.pps (демонстрація), то це дасть суттєві переваги у її використанні. ЕСУ можна буде управляти без послуг *Microsoft PowerPoint*, як звичайним додатком *Windows*. ЕСУ слід поєднувати з використанням КЕДЗ і прикладних пакетів.

В зв'язку з тим, що учні переважну частину уроку з використанням ЕСУ індивідуально працюють з комп'ютером, він завжди має елементи застосування ІКТ, але здебільшого цього буває недостатньо, тому ЕСУ корисно застосовувати на бінарних уроках¹².

Для оцінювання якості ЕСУ також необхідні певні критерії, тому наведемо та коротко прокоментуємо зразок:

таблиця II.2

№	критерій	оцінка в балах
1	Відповідність темі уроку	___ балів
2	Відповідність загальним вимогам до електронних навчальних засобів	___ балів
3	Структура ядра (головної презентації)	___ балів
3.1	Якість головного меню (плану)	___ балів
3.2	Ступінь відображення плану	___ балів
4	Якість інтерфейсу (навігаційні можливості)	___ балів
4.1	Кнопки переходів	___ балів

¹² Різницю уроків з використанням ІКТ та бінарних уроків з інформатикою розглянемо нижче.

4.2	Підказки	__ балів
5	Змістовність слайдів	__ балів
6	Якість унаочнення	__ балів
6.1	Якість графічних елементів	__ балів
6.2	Якість анімаційних елементів	__ балів
7	Інтеграційні можливості	__ балів
8	Коефіцієнт змістовності	__ балів
9	Можливості редагування	__ балів
10	Можливості закріплення	__ балів
	Загальна кількість балів	__ балів

Порівняння орієнтовних критеріїв оцінювання ЕКУ та ЕСУ приводить до висновків: по-перше, спостерігається зовнішня та деяка змістовна подібність, по-друге, існують суттєві й принципові відмінності в підході до оцінювання. Зовнішня, а особливо змістовна подібність пояснюється тим, що обидві форми ЕДЗ – це, як уже зазначалось вище, засоби ІКТ, тобто стосуються форми представлення та вивчення навчального матеріалу. Відмінності випливають із принципової відмінності ЕКУ та ЕСУ (нагадаємо, ЕКУ – це поєднання методичних поміток для викладача та стислого конспекту для учнів, а ЕСУ – це спільна для учителя та учнів алгоритмічна схема заняття).

При оцінці ЕСУ перш за все слід виявити відповідність темі уроку (п.1) та загальним вимогам до електронних навчальних засобів (п.2). Проте, на відміну від ЕКУ може охоплювати лише один урок, або одну вузьку тему, що й вимагається в п.1.

Оцінка якості структури ядра, тобто головної презентації (п.3), дуже важлива, вона функціональність та якість головного меню, а також якість та ступінь наповнення ядра відповідно плану.

ЕКУ - переважно цілісні текстові документи, вони не мають інтерфейсу, навігаційні можливості здійснюються практично виключно з допомогою гіперпосилань. Для ЕСУ якість інтерфейсу має суттєве значення, адже відповідно до специфіки він потребує більш чітко виражених і надійних навігаційних можливостей. Тому п.4 включає аналіз системи кнопок переходів та впливаючих підказок.

Якість будь-якої презентації залежить від якості окремих слайдів. Для ЕСУ ця обставина особливо важлива, адже ми маємо справу з презентацією для індивідуальної, а не колективної роботи. Звідси впливає необхідність послайдового та покадрового¹³ контролю за змістовністю, наочністю та естетичністю ілюстративного матеріалу (п.5-6).

Пункти п.7, 9-10, які присутні й серед критеріїв оцінювання ЕКУ, але неважко зрозуміти, що при оцінюванні ЕСУ вони мають трохи змінені акценти. Наприклад, п.10 не потребує такої деталізації, як це видно для випадку ЕКУ. Особливо слід вказати на п.8. Він введений для контролю за тим, щоб досягалось оптимальне співвідношення між кількістю слайдів

1.6 ПРИКЛАД ЕЛЕКТРОННОГО СЦЕНАРІЮ УРОКУ

¹³ Під кадрами будемо розуміти модифікації слайдів, досягнену з допомогою анімаційних ефектів.

Особливістю уроків з ЕСУ є універсальний характер по відношенню до навчальних предметів. У зв'язку з цим за приклад візьмемо урок алгебри на тему «Застосування логарифмічної функції»¹⁴. Спочатку опишемо ядро. Кожен слайд пронумеровано в нижньому лівому кутку. Отже ядро складається з 8 слайдів, перший з них містить головне меню. Навігація по ядру реалізована через гіперпосилання на слайди, про що свідчить статична підказка «Виберіть пункт та клацніть його» та стандартні кнопки НАЗАД і ДОВІДКА.

Як видно з ілюстрації до другого слайду він містить два кадри, причому, перший кадр (зліва) треба вважати *базовим*, а другий *похідним*, бо він (справа) одержується з першого додаванням завдань та підказки для їх виконання.

Третій слайд, крім базового, містить цілу ієрархію похідних, з них вісім лише є похідними першого рівня (бо одержуються з базового одним доповненням). Створення таких кадрів досягається з допомогою вставки анімаційних ефектів та перемикачів. Доцільність таких дій очевидна, адже без них довелося би створювати надто багато слайдів, що привело б до надмір-

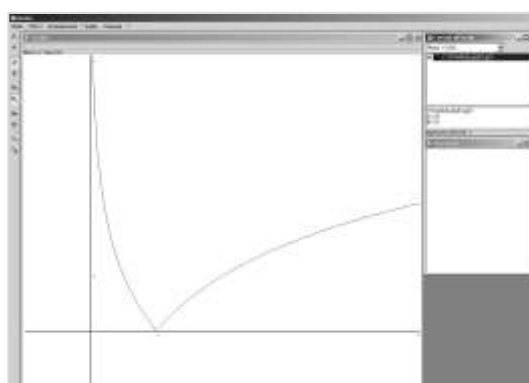
¹⁴ Без ілюстрації більшості слайдів важко стисло описати ЕСУ в цілому, але умови друку дозволяють їх представляти у чорно-білому варіанті.

ності розміру файлу, який містить презентацію, а їх лінійна демонстрація мала б дуже малу дидактичну цінність.

Багаторівневу ієрархію слайду легше проілюструвати на «біднішому» анімаційні елементами та навігаційними можливостями четвертому слайді, бо він всього три похідні кадри. Кожен, хоч трохи обізнаний з тех-

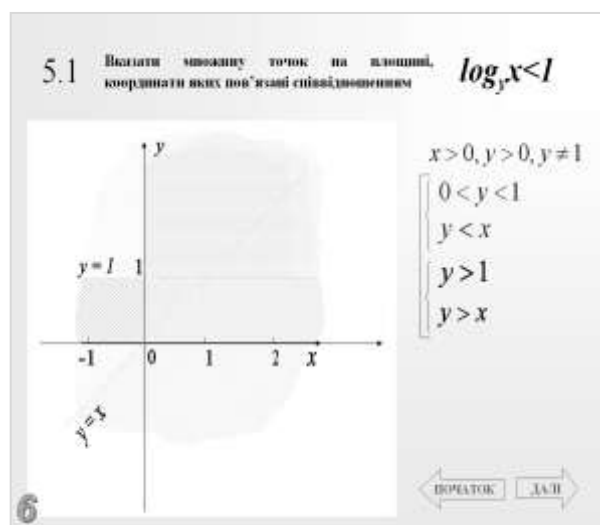
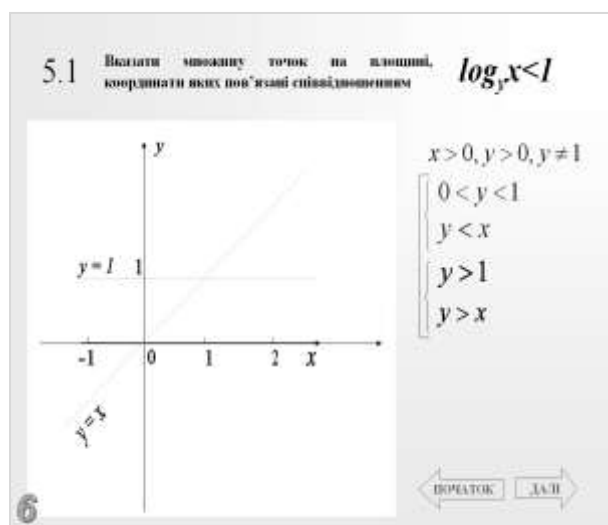


нологією роботи з *Microsoft Power Point*, знає, що можна досягти практично

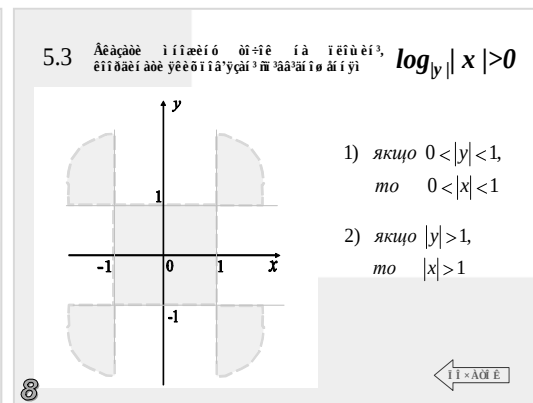
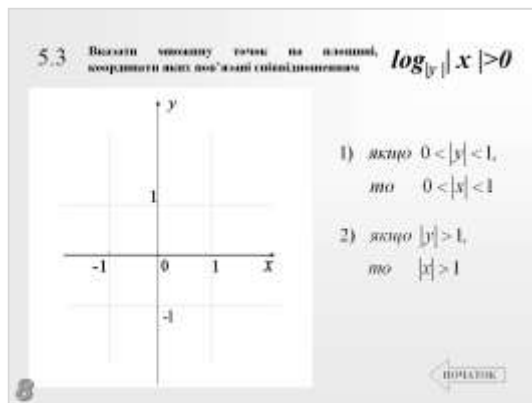
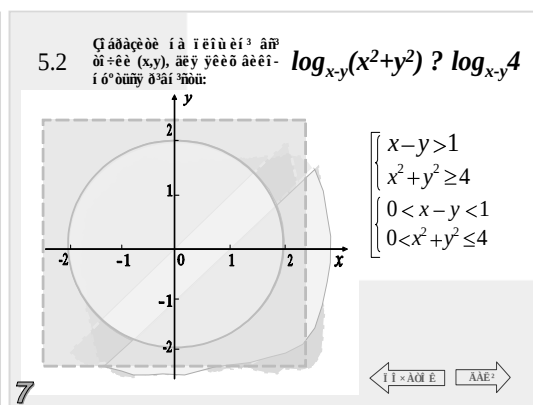
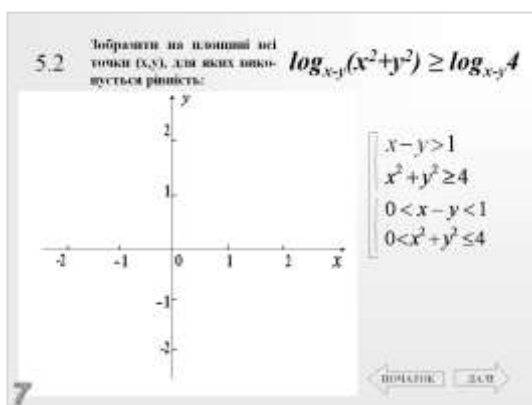


будь-якої послідовності демонстрації слайдів та кадрів. Такі широкі навігаційні можливості пакету створюють надзвичайно потужні й цінні дидактичні можливості.

Окрему роль має п'ятий слайд, що містить єдиний кадр. Він призначений для переходу в автономний тренувальний режим (схематична побудова графіків функцій у робочих зошитах та перехресна перевірка з допомогою пакета *Gran1*).



Для скорочення опису всього ядра далі наведемо лише базовий та вихідний (остаточний) кадри наступних слайдів (№ 6-8), сподіваючись, що



уважний читач зможе уявити досить точно все ядро ЕСУ.

Для слайдів 1 та 6-8 пропонуємо самостійно провести аналіз навігаційних та дидактичних можливостей, як це було зроблено вище для слайду 4.

Залишається нагадати, що при потребі легко провести необхідне редагування або змінити інтеграційні зв'язки з іншими додатками з допомогою гіперпосилань чи кнопок переходів, як це було видно у слайді 5 відносно додатка *Gran1*. Для цього необхідно відкрити файл, що містить ядро, у форматі *.ppt.

Пропонуємо оцінити описаний приклад ЕСУ з допомогою наведених критеріїв оцінювання.

РОЗДІЛ II.

ЗАСТОСУВАННЯ МЕТОДУ ПРОЕКТІВ У ПРОЦЕСІ ВИКЛАДАННЯ ОСНОВ ІНФОРМАТИКИ.

Вище було вказано на роль і місце інноваційних технологій навчання в сучасній освіті. Проте ще зовсім мало згадувались інтерактивні методи навчання, які мають пряме відношення до ІТН.

Термін «інтерактивний» має англійське походження і означає «взаємодіючий». Існують різні підходи до визначення інтерактивного навчання, зокрема, його часто трактують, як діалогове навчання: «Інтерактивний – означає здатність взаємодіяти чи знаходитись в режимі бесіди, діалогу з чим-небудь (наприклад, комп'ютером) або ким-небудь (людиною). Отже, інтерактивне навчання – це перш за все діалогове навчання, в ході якого

здійснюється взаємодія вчителя та учня» [13]. За О.Пометун та Л.Пироженко «Сутність інтерактивного навчання полягає в тому, що навчальний процес відбувається за умов постійної, активної взаємодії всіх учнів. Це співнавчання, взаємонавчання (колективне, групове навчання в співпраці)...»[14].

Розшифровуючи поняття «інтерактивні методи» буквально, можна погодитись, що їх зміст походить від двох складових: «інтер» та «активний», тобто означає методи групової активізації іззовні. Звідси випливає справжня, а не формальна сутність інтерактивних методів навчання. І вона означає *колективне добування знань через зовнішню керованість*, тобто через *маніпуляцію свідомістю об'єкта навчання*¹⁵.

Тому інтерактивні технології не можуть бути самоціллю [16, 17]. Потрібно постійно контролювати процес досягнення поставлених цілей (вони повинні бути чітко сформульовані і легко контрольовані), у випадку невдачі переглядати стратегію і тактику роботи, шукати і виправляти недоліки. *Урок ні в якому випадку не повинен бути перевантаженим інтерактивною роботою*. Оптимально рекомендується застосовувати не більше 1-2 методів за урок. Слід поєднувати взаємонавчання з іншими методами роботи, зокрема з традиційними методами та *самостійним пошуком*. Неможливо побудувати весь процес навчання виключно на інтерактивних методах. Це один з багатьох прийомів, які допомагають досягнути мети і приносять результат тільки в поєднанні з іншими.

Одним із найвідоміших та найефективніших інтерактивних методів вважається метод проектів. Не слід плутати *метод проектів* - як форму організації навчально-виховного процесу та *проект* - як етап розв'язання проблеми. Сьогодні більше акцентують увагу саме на методі проектів, маючи на увазі одну із форм інтерактивних технологій. Але – це класичний приклад того, як за лісом не бачать дерев, або, захопившись вихованням, гублять його об'єкт – дитину. Отже, без правильного розуміння поняття “проект” не можна організувати ні проектну діяльність, ні застосувати метод проектів.

Перед тим, як сконцентруватись на поняттях «проект» та «проектна діяльність» корисно згадати, що існує класична, але по суті абсурдна класифікація видів навчання. Говорять, що існує *пасивне, активне та інтерактивне* навчання. З цим ніяк не можна погодитись. Хіба не безглуздо стверджувати, що існує пасивне навчання? *Навчання – це процес, в якому одночасно і активно приймають участь або дві особи – учитель і учень, або одна особа - учень*. Якщо учень пасивний (читай - відсутній), то це зовсім не навчання, причому, можна сказати «за визначенням». Отже, **будь-яке навчання не буває пасивним**. Щодо інтерактивного навчання можна сказати наступне. Інтерактивне – значить стимульоване із зовні, тобто навчання під впливом зовнішніх стимуляторів. Тому інтерактивні елементи в навчально-виховному процесі корисно застосовувати *не постійно, а епізодично*, наприклад, для актуалізації, концентрації, закріплення чи в процесі повторення або узагаль-

¹⁵ Тут термін «маніпулювання свідомістю» вжито у вузькому, пов'язаному з навчально-виховним процесом, розумінні, його дія не пов'язується із формуванням особистості та суспільною поведінкою.

нення. Слід завжди мати на увазі, що *інтерактивні методи* – це окремий вид *маніпуляцій свідомістю*, їх надмірне «вживання» може привести до формування в учнів ефекту (а то й дефекту) залежності, тобто нездатності самостійно мислити, виконувати роботу, приймати рішення.

Важко погодитись, що вже згадана широко відома програма «Інтел® Навчання для майбутнього» правильно орієнтує вчителів щодо підвищення якості навчально-виховного процесу. Основна вада цієї програми в тому, що під керівництвом учителя учні повинні вибирати так звані *ключове* та *тематичні питання*, що стосуються проблеми, яку *тільки збираються досліджувати*. Не знаючи предмету дослідження, учні не можуть свідомо, без допомоги учителя, вибирати жодних конкретних питань. *Проект по своїй суті – це алгоритм розв'язування поставленої задачі*, а не сукупність прийомів досягнення результатів. Ми часто спостерігаємо, як при формулюванні алгоритма чи опису програми учні концентрують увагу не на *визначенні суми, пошуці, сортуванні, переборі* тощо, а говорять про *цикли, масиви* і т.д. Цикли – це лише інструменти, що можуть допомогти знайти суму, виконати пошук, сортування, перебір і т.д., а масиви – це форма представлення даних. Самі ж питання, як *тематичні, так і ключові*, можуть виникати в процесі розвитку (реалізації) проекту. Ніякого відношення до проекту не мають також різноманітні інтерактивні прийоми, адже, як зазначалось вище, їх суть в маніпулюванні свідомістю учнів, нехай і в позитивному розумінні, тоді, як проектна діяльність повинна бути свідомим вибором кожного учня.

Спробуємо дати відповідь на питання: *чи можливо навчити?* Відповідь: неможливо, але *можливо навчитись*. Тоді виникне питання: Як учитись і навчитись? Відповідь: *учитись* – це самостійно *шукати запитання і відповіді на них*. З огляду на це можна говорити про навчально-пізнавальний проект, який має на меті освіту. Він полягає в організації навчальної діяльності, за принципом: *від керованого і контрольованого до самостійного навчання*.

Проект - у буквальному значенні - це «кинутий вперед», тобто прототип якогось виду діяльності. Виконавець будь-якого завдання, якщо він достатньо організований, весь час процює з проектом - спочатку планує (*проектує*) майбутню роботу, потім поетапно *виконує проект*. Отже проект може бути в двох стадіях: стадії *формулювання* і стадії *виконання*. Зрозуміло, що спочатку слід навчитись виконувати проекти, запропоновані кимось іншим. У цьому випадку складання проекту і контроль за його виконанням покладається на *керівника проекту*. Як правило, до цього зводяться традиційні форми навчальної роботи.

У педагогічній літературі наводиться кілька типів проектів, що використовуються у шкільному навчанні.

1. *Дослідницькі проекти*, які цілком підпорядковані логіці дослідження і мають структуру, наближену або повністю відповідну справжньому науковому дослідженню (аргументація, актуальність теми, визначення проблеми, предмета і об'єкта дослідження, окреслення завдань, методів дослідження, джерел інформації, висування гіпотез, означення шляхів її розв'язання, обговорення отриманих результатів, їх оформлення).

2. *Творчі проекти*, які не мають детально опрацьованої структури діяльності учасників. Вона лише накреслюється і розвивається відповідно до жанру і форми кінцевого результату (спільна газета, твір, відеофільм, драматизація тощо.)

3. *Пригодницькі та ігрові проекти*, які також не мають чіткої структури. Вона визначається під час опрацювання учасниками проекту ролей, зумовлених характером і змістом проекту (історичні особи, вигадані герої, які імітують офіційні та ділові відносини).

4. *Інформаційні проекти*, які зорієнтовані на збирання інформації, мають чітку структуру (мета, актуальність, методи отримання і обробки інформації, оформлення результатів та їх презентація).

5. *Практично-організаційні проекти*, спрямовані на вироблення конкретної програми дій, методичних рекомендацій.

Методика виконання проектів передбачає:

- Вибір теми або проблеми. Темі може запропонувати вчитель або ж їх вибирає клас безпосередньо із тих пропозицій, які були запропоновані під час «мозкової атаки».
- Планування проекту. Вчитель і учні узгоджують всі складові процесу виконання проекту (на який час він розрахований, які ресурси будуть використані, яким чином учні працюватимуть (групою чи поодиночі).
- Процес дослідження / дії.
- Результати і / чи висновки.
- Представлення результату роботи.
- Оцінка результатів і процесу.

Проект може бути монопредметним, міжпредметним і надпредметним. Виконання проектів має різні етапи, однак на всіх етапах необхідно наголошувати, що відповідальність учнів за їхнє навчання лежить на них самих.

Це дуже ефективний спосіб співпраці у рамках узгодженої теми чи питання. Він вчить окреслювати цілі, планувати (структурувати) роботу, оцінювати результати. Передбачає високу активність і незалежність виконавців проекту.

Процес навчання має різні рівні, найвищим з яких є *самостійне навчання*. Для цього рівня необхідні достатні навички як виконання проектів, так і їх складання, тобто *проекткування*. При самостійному навчанні метод проектів має зовсім інший зміст, ніж при застосуванні інтерактивних технологій.

Третє заняття ОШППД (2008 р.) було присвячене застосуванню методу проектів. Розглядалися два варіанти застосування цього методу: для підготовки та проведення уроків-семінарів та при організації державної підсумкової атестації у формі захисту навчальних проектів.

Одна із ефективних структурних схем семінарської роботи така:

1. План підготовки до семінару;
2. План проведення семінару.

На першому етапі слід встановити тему та ключове питання семінару, терміни підготовки, розподілити теми та тематичні питання для учасників, питання для всіх учасників, пов'язані з теоретичними положеннями вибраної теми, форми підготовки питань (реферати, виступи, повідомлення тощо), форми роботи над питаннями (індивідуальні, групові, колективні).

На другому етапі слід визначити послідовність проведення семінару, вибрати спікерів, призначити доповідачів, асистентів, опонентів, встановити регламент, систему балів для оцінювання, форми опитування та підведення підсумків.

Подібну схему може мати й захисту проектів під час державної підсумкової атестації з інформатики. Різниця в тому, що тут слід строго дотримуватись відповідної інструкції МОН України. Але процес підготовки державної підсумкової атестації з інформатики у формі захисту проектів має чисто індивідуальний характер, учень працює самостійно, але під наглядом учителя.

Вище було достатньо переконливо обґрунтовано, що групові форми роботи можуть використовуватись лише епізодично, *найефективнішими є самостійні форми навчальної діяльності*. При будь-якій нагоді їм слід надавати перевагу, а випускники загальноосвітньої школи повинні бути повністю готовими добувати знання та інтелектуально удосконалюватись самостійно.

Під час одного із занять ОШППД слухачі спостерігали, як учень працював над проектом, який можна назвати «Розв'язування олімпіадної задачі з програмування». Він складав загальний план: оголошував та описував відповідно до умови задачі дані й шукані величини, обдумував математичну модель задачі, описував, уточнював, доводив та оцінював алгоритм, писав та відлагоджував програму, для чого придумував відповідні умови задачі тести, нарешті виконував програму з метою одержання результатів. Робота над задачею з програмування – це чудовий зразок *самостійної проектної діяльності*. Обґрунтування такого погляду постійно зустрічається в [7]-[12].

РОЗДІЛ III

III.1 ВИВЧЕННЯ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ У ФОРМІ ЕКСПРЕС-КУРСУ. ПРОБЛЕМИ ВИВЧЕННЯ ТЕМИ «КОМП'ЮТЕРНЕ МОДЕЛЮВАННЯ. ОСНОВИ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ» У ШКОЛІ

Серед семи найактуальніших проблем сучасної шкільної інформатики перші дві вказують на труднощі, що виникають при вивченні теми «Комп'ютерне моделювання. Основи алгоритмізації та програмування». Враховуючи, те, що логічно ця тема стоїть окремо і залишається однією з ключових у всьому курсі, виникла необхідність окремо виділити і деталізувати проблеми її вивчення [6].

Цілком слушно вважати названу тему *розділом*, адже протягом багатьох років весь курс інформатики майже виключно зводився до вивчення алгоритмів і програм. Крім того, *темою* прийнято називати частину деякого розділу певного предмету чи курсу, тоді, як «Алгоритмізація і програмування» логічно не належить до жодної з частин курсу інформатики, які прийнято

скорочено називати АС, СПЗ чи ППЗ¹⁶ тощо, а враховуючи логічну відособленість від решти навчального матеріалу шкільної інформатики, цю тему доцільно вважати навіть *окремим курсом*. Щоправда, курс, як правило, має значно більшу кількість навчальних годин.

Вже з назви цього розділу чи курсу, що поєднує терміни *комп'ютерне моделювання, основи, алгоритмізація та програмування* зрозуміло, що його зміст належить не до предметів циклу «технології», а дуже близький до математичних дисциплін, тому повинен бути точним, конкретним і стислим. Це цілком підтверджується відведеною у програмі кількістю навчальних годин (12 – 17,1% від загальної кількості навчальних годин)¹⁷. Якщо врахувати, що у переважній більшості загальноосвітніх навчальних закладів інформатика вивчається у обсязі універсального профілю, при його викладанні практично перед усіма вчителями виникає багато особливо гострих питань і проблем, які дуже важко вирішити. Розглянемо деякі з них, що, на наш погляд, цілком доречно поставити:

- Який сенс вивчати у школі такий досить абстрактний і формалізований розділ *оглядово*?
- Як «втиснути» цей розділ у 12 навчальних годин?
- Чи можливо при цьому *досягти* поставленої мети та засвоєння учнями вказаного рівня теоретичних знань і практичних навичок?

Виходячи з того, що учителі не в змозі змінювати програму, ці питання можна вважати риторичними. Але тоді проблема, пов'язана з ними залишиться. Її можна сформулювати так: ***Проблема забезпечення необхідного рівня теоретичних знань і практичних навичок при викладанні розділу курсу основ інформатики «Комп'ютерне моделювання. Основи алгоритмізації та програмування» згідно діючої програми в умовах недостатньої кількості відведених навчальних годин.***

III.2 МЕТОДИЧНЕ ОБҐРУНТУВАННЯ ЕКСПРЕС-КУРСУ АЛГОРИТМІЗАЦІЇ ТА ПРОГРАМУВАННЯ

Перш за все слід визначити достатньо ефективний шлях вирішення поставленої проблеми. Практика вивчення алгоритмізації та програмування підказує, що для ефективності необхідно забезпечити єдність *формування загальних уявлень про структурне програмування і послідовності деталізації переходів від моделі задачі, до алгоритму і програми*. Безперечно, розв'язати це завдання можна лише шляхом *максимального ущільнення* навчального матеріалу, *універсалізації* задач, запропонованих для його закріплення, забезпечення *логічної єдності* всього навчального матеріалу та при умові *обов'язкового виконання* учнями *теоретичних і практичних завдань у позаурочний час*. ЕКАП повинен формуватись із кількох послідовних динамічних модулів. Необхідно передбачити можливість доповнення кожного модуля,

¹⁶ Маються на увазі частини обчислювальної системи: АС – апаратна складова, СПЗ – системне програмне забезпечення, ППЗ – прикладне програмне забезпечення.

¹⁷ На даний час у зв'язку із введенням рівнів вивчення ситуація дещо змінилась: тема практично відсутня для рівня стандарту (7,1% від загальної кількості навчальних годин) і трохи розширена (28 навчальних годин або 26,6%).

щоб він, не втрачаючи цілісності, допускав розширення або стиснення. Не слід відмовлятися від будь-якої нагоди розширити ЕКАП за рахунок ущільнення інших тем курсу, яке, виходячи з місцевих умов, допускається авторами програми.

Зрозуміло, що при застосуванні до розділу таких умов його справді можна називати *експрес-курсом з основ алгоритмізації та програмування* (ЕКАП) для загальноосвітніх шкіл згідно діючих навчальних планів та програм для академічного рівня, який можна прийняти за базовий¹⁸. Термін «експрес» тут вжито як в традиційному розумінні, тобто «прискорений», адже програмування за 12-15 навчальних годин – це справді надзвичайно швидко і стисло, так і в розумінні «особливий», «надзвичайний» та «ефективний». Щоб зробити ЕКАП більш універсальним, його можна доповнити серією з 6-8 практичних робіт. Таким чином для базового складу ЕКАП необхідно передбачити 18-25 уроків, які разом з двома-трьома заліковими уроками заповнять передбачену навчальним планом кількість навчальних годин.

Виникає природне питання про відповідність ЕКАП-у програмі з інформатики. Він не може повторювати теми та уроки, взяті з програми. Чи допустимо це? По інструкції – ні, але виходячи із здорового глузду – так. В умовах, коли циклічно, кожні 5-7 років повністю міняються і програми і підручники, треба вважати пріоритетним формування цілісного уявлення про алгоритмічне й структурне мислення в допустимому обсязі.

Не маючи можливості навести ЕКАП повністю, відсилаємо читачів до [6], у додатку 2 пропонується познайомитись із серією практичних робіт, які логічно завершують ЕКАП. Кожен учитель при потребі та бажанні матиме можливість, врахувавши ці методичні поради, самостійно створити курс лекцій і практичних занять, більш чи менш стислий і компактний.

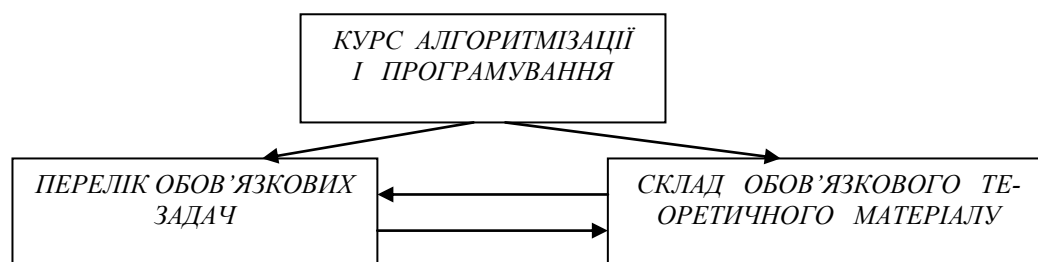
Створити такий ЕКАП не можливо без застосування певних *робочих ідей*, або *принципів*, які покликані допомогти у досягненні ущільнення, універсалізації, логічної єдності і т.д. Тому дуже доречно згадати думку Р.Декарта: «Кожна розв'язана мною задача ставала образом, який застосовувався надалі при розв'язуванні інших задач». Вона висловлює загальновідому, навіть банальну в наш час дидактичну істину, але все геніальне насправді просте. Дійсно, курс обов'язково передбачає розгляд певної кількості задач, розв'язання кожної з яких перш за все потребує математичної постановки. Учням, що в процесі вивчення розділу алгоритмізації та програмування ще тільки знайомляться з новими різновидами мислення, зокрема *алгоритмічним* та *структурним*, при цьому буде необхідно не тільки багато додаткового часу, а й значних інтелектуальних зусиль, спрямованих на подолання психологічних бар'єрів у вигляді постійної новизни і невідомості та розпорошення уваги, яка приводить до втрати почуття цільності курсу. Тому будемо намагатись розглядати таку *серію задач, де кожна наступна - в певному розумінні буде розширенням чи аналогією попередньої*, іншими словами, з метою економії часу та

¹⁸ Описані тут ідеї та методи можна адаптувати і до інших варіантів діючих програм з інформатики для загальноосвітніх навчальних закладів.

інтелектуальної напруги спробуємо реалізувати ідею викладу курсу програмування на одній великій задачі, або за **принципом нитки** (умовно позначимо його літерою **A**).

Будь-який курс повинен охоплювати певне завершене коло основних задач, матимемо це на увазі й при створенні нашого ЕКАПу. Перелік задач повинен відповідати діючій програмі. Крім того, виконуючи її, ми повинні забезпечити необхідний рівень засвоєння мінімального обсягу теоретичного

схема III. 1



матеріалу, тобто необхідно скласти взаємопов'язані повний перелік обов'язкових задач і мінімальний обсяг теоретичного матеріалу (схема IV.1).

Наприклад, треба розглянути задачі:

		список 1
номер та назва обов'язкові задачі		номер відповідного задачі теоретичного матеріалу
1	Написати лінійну програму обчислювального характеру	1,2,3,4,6
2	Написати діалогову програму	1,2,3,4,5,6
3	Написати програму із розгалуженням	1,2,3,4,5,6,8
4	Написати циклічну програму	1,2,3,4,5,6,7
5	Написати програму визначення суми заданих формулою чисел	1,2,3,4,5,6,7
6	Написати програму визначення суми даних чисел, заданих масивом	1,2,3,4,5,6,7,9,10
7	Написати програму пошуку в лінійних неупорядкованих числових масивах	1,2,3,4,5,6,7,8,9,10)
8	Написати програму пошуку в лінійних упорядкованих числових масивах	1,2,3,4,5,6,7,8,9,10)
9	Написати програму сортування масивів	1,2,3,4,5,6,7,8,9,10,11

У якості відповідного цим задачам обов'язкового теоретичного матеріалу можна взяти теми:

список 2

1. Поняття величини в інформатиці, оголошення і опис величин;
2. Математичні операції та функції над числовими типами та запис математичних виразів у лінійній формі;
3. Команда присвоєння;
4. Процедури вводу/виводу;
5. Символьний та рядковий типи;
6. Коментарі в програмі;
7. Поняття циклу;
8. Поняття команди розгалуження;

9. Поняття масиву;
10. Ввод масиву;
11. Вивод масиву.

Як бачимо, пункти першого списку слід узгодити при вивченні із деякими пунктами другого списку, причому, з кожним наступним ідуть повтори та наростання обсягу обов'язкового теоретичного матеріалу (у першому списку відповідні пункти другого списку наведені справа в дужках). Таким чином, що цілком природньо, застосування однієї і тієї ж порції теоретичного матеріалу при розгляді різних задач часто повторюється у різних комбінаціях з іншими порціями, а нерідко дублюється (звернемо увагу хоча б на порції теоретичного матеріалу 1-4, які дублюються в усіх дев'яти наведених задачах). Скориставшись щойно сказаним, можна досягти значного ущільнення як навчального матеріалу так і часу, відведеного для вивчення. Як видно із списку 1, порції теоретичного матеріалу з меншими номерами на кожному наступному кроці викреслюються, наприклад, починаючи із задачі 3, викреслені п.1-4, починаючи із задачі 7 викреслені п.1-7 і т.д. Це слід розуміти так: викреслені у списку 1 порції теоретичного матеріалу з моменту викреслення більше не вивчаються і не повторюються, наприклад, перед розглядом задачі 8, можна обійтись вивченням тільки двох порцій теоретичного матеріалу: поняття команди розгалуження (№8) та ввод масиву (№10), а з рештою тем (№№1-7 та №9) ознайомлення відбулось при розв'язуванні попередніх семи задач.

Але чи можна обійтись у цій ситуації навіть без повторення тем 1-7 та 9? Відповімо: *не тільки можна*, а при вивченні розділу у вигляді ЕКАПу і *потрібно*. Тут ми свідомо йдемо проти вікової дидактичної, але, на жаль, в дійсності *не зовсім педагогічної традиції* «повторення – мати навчання». Чому практично на всіх уроках організовується повторення попереднього навчального матеріалу? Щоб створити максимум комфорту для учнів. Але погодимось, що безкінечне повторення прописних істин – це зовсім не цікаво для тих учнів, які давно засвоїли не тільки напам'ять а й до рівня розуміння. Скільки таких учнів у класі з n осіб? Якщо цю величину позначити k , то ось строга відповідь: залежно від ситуації $0 \leq k \leq n$. Напрошується цікавий висновок: *у навчальному процесі не доцільно зловживати повторенням раніше вивченого*, інакше в учнів можуть виробитись неухажність, байдужість, бездіяльність, безініціативність і т.д., замість повсякденного планування повторення *слід постійно слідкувати, щоб k було якомога ближче до n , а на деякому етапі навчання досягало його*.

Справді, чи завжди у повсякденному житті починають роботу з повторення, тренування тощо? Це доцільно, наприклад, спортсменам, співакам тощо, для яких так звана розминка – необхідність. Для більшості ж видів людської діяльності *основні прийоми та навички повинні бути доведені до автоматизму*. Щоб переконатись у цьому, візьмемо будь-яку масову професію, хоча б столяра. Хіба він, починаючи робочий день, повторює навички пиляння, стругання, забивання цвяхів і т.д.? Отже, в ідеалі процес навчання повинен забезпечувати на будь-якому етапі абсолютне засвоєння необхідних

знань, умінь та навичок, тому, на нашу думку, вчителям слід організовувати навчальний процес якомога ближче до ідеального.

Поміркуємо, як цього досягти. Відповідь можна знайти в учителів-новаторів, наприклад у І. П. Волкова¹⁹. Під впливом одного із його виступів ще в першій половині восьмидесятих років ХХ століття закарбувалась чітка схема розвитку творчих здібностей та виявлення професійних нахилів учнів:

- 1) вивчаючи будь-який курс, предмет чи окрему тему слід виявити *основні поняття та їх властивості* (у І. П. Волкова, наприклад, інструменти у майстерні по дереву та основні навички роботи з ними);
- 2) досконало вивчити виявлені основні (робочі) поняття та до автоматизму відпрацювати стандартні прийоми їх використання (у майстерні – це правила техніки безпеки при використанні ножа, пилки, стамески тощо та стандартні прийоми їх застосування при роботі з деревом);
- 3) відпрацювати комплексне застосування робочих понять (наприклад, одночасне використання стамески і молотка);
- 4) створити умови для *самостійної творчої роботи* у рамках теми з вибраними поняттями та їх властивостями (створення виробу із дерева).

По суті, ми підійшли до другого робочого принципу (Б), необхідного для створення ЕКАПу. Коротко його можна сформулювати так: ***забезпечувати засвоєння навчального матеріалу не на рівні запам'ятання а на рівні розуміння, щоб після закріплення зникла потреба в повторенні раніше вивченого.***

На користь цього принципу наведемо досить тривіальний приклад. Вчителі інформатики знають, як важко призвичаїти учнів до строгості мови програмування, застерегти від багаторазового повторення банальних помилок порушення її синтаксису, дотримання структури програми тощо. Візьмемо фрагмент довільної програми, що містить блок вводу величини *a* з виводом підказки:

```
{...} var a:real; {...}  
begin  
  {...} Write('a?');Read(a); {...}  
end.
```

Можна спостерігати, як у подібних ситуаціях окремі учні на протязі багатьох уроків набирають “різні варіанти”:

- 1) Write(a?);Read(a);
- 2) Write('a? ');Read('a');
- 3) Write('a? ') Read(a) тощо.

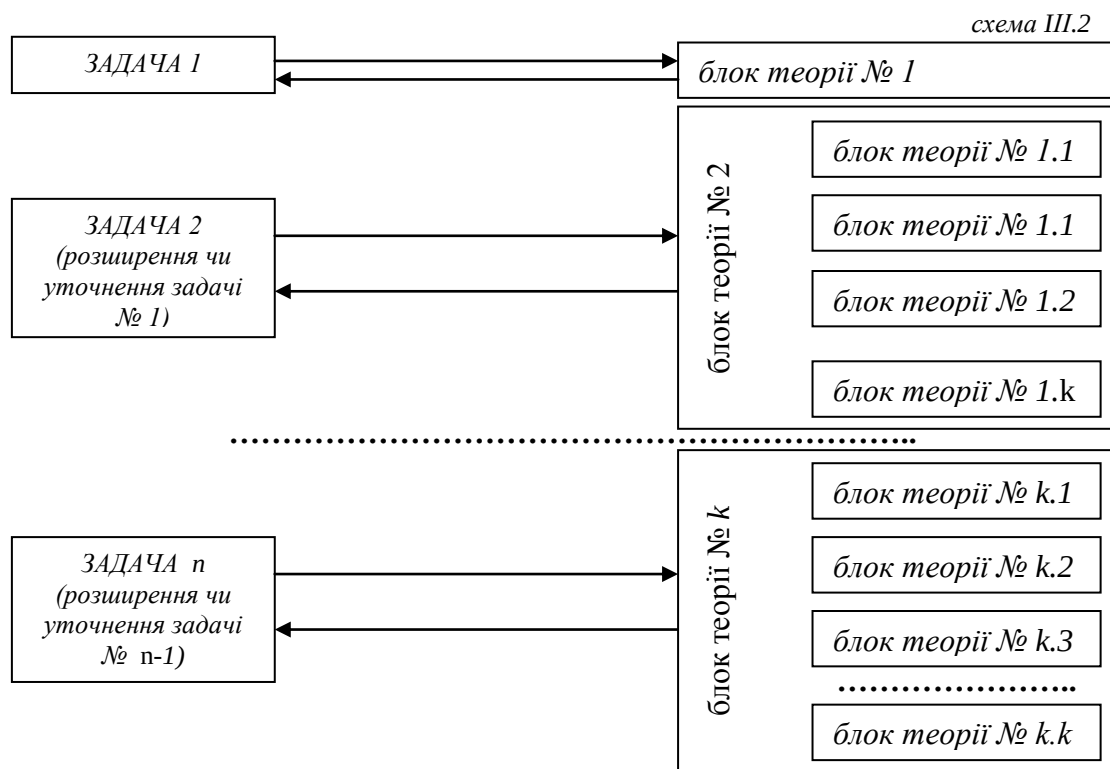
Як подолати раз і назавжди допущення таких помилок, щоб не тратити час на кожному уроці на повторення формату та різних варіантів команд

¹⁹ І. П. Волков, учитель малювання та праці з м. Реутово Московської області, кандидат педагогічних наук, заслужений учитель школи РСФСР. Його основна новаторська ідея - створення механізму реалізації творчих здібностей дітей і здійснення педагогічного керівництва процесом розвитку індивідуальності кожного школяра.

вводу і виводу? Дуже просто. Дати учням можливість «набити» усі «шишки» відразу, виконавши наведені варіанти в середовищі системи програмування. Учень повинен *усвідомити*, а не *запам'ятати* правильний запис. Для цього у кожного є свій ліміт повторень ситуацій. Очевидно, що краще цей ліміт вичерпати відразу. Читачеві нічого не залишається, як погодитись, що після використання описаного дидактичного прийому важко буде знайти учня, якому для уникнення помилок необхідне повторення.

Усе це дасть значну економію часу і дозволить навчати без надмірних інтелектуальних напружень. А наведена щойно схема розвитку творчих здібностей учнів зовсім не є інноваційною, вона лише вимагає будувати ЕКАП по строгості й логічності подібно аксіоматичному методу побудови курсу геометрії.

Безперечно, на практиці впровадження принципів **А** та **Б** не таке прозоре і зрозуміле, тому додатково скористаємось схемами 2 та 3.



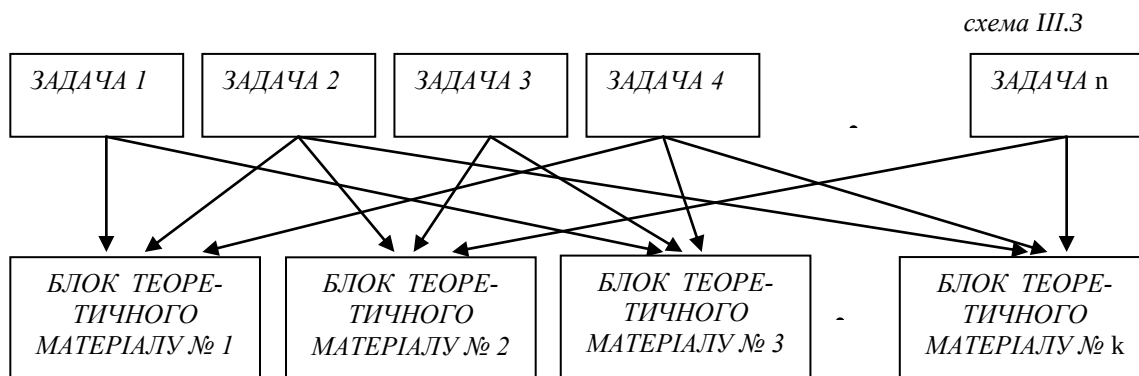
Якщо при вивченні алгоритмізації і програмування дотримуватись наведеної вище схеми III.2, то можна зняти психологічні складності, отже підвищити якість засвоєння навчального матеріалу. Звичайно, при підборі задач необхідно уявляти весь курс і починати із останньої задачі. Нехай це *задача n* про пошук у відсортованому лінійному числовому масиві. Логічно найближча до неї задача про сортування лінійного числового масиву, тому назвемо її *задачею n-1* і будемо вважати попередньою. Для задачі *n-1* попередньою може бути задача *n-2* про відшукування найменшого елементу в лінійному числовому масиві і т.д. Міркуючи таким чином, поступово можна дійти до задачі 1. Але задачу *n* можна сформулювати настільки детально, що вона міститиме всі попередні - від задачі *n-1* до першої включно. На кожному етапі деталіза-

ції чи ускладнення поточної задачі можна підключати необхідний блок теоретичного матеріалу.

В решті-решт весь курс буде присвячений розгляду однієї глобальної задачі з поступовим нарощуванням блоку теоретичного матеріалу, як і передбачено принципом А (див. схему ІІІ.2).

Застосовувати А можна лише умовно, адже постійно доповнюючи умову задачі, ми одержуватимемо інші задачі (наприклад, із списку 1). Крім того, можна укрупнити ЕКАП, використавши не одну, а кілька базових задач, які не зводяться одна до одної. Уявимо, що в курсі їх буде n . Звернемо увагу на те, що і блоки теоретичного матеріалу тоді в задачах будуть застосовуватись не так, як показано у списку 1, а більш хаотично (див. схему ІІІ.3).

Як уже зазначалось, для забезпечення вивчення алгоритмізації та програмування в прискореному і ефективному режимі, тобто режимі експрес-курсу, необхідні додаткові резерви часу і обов'язкова організація самостійного виконання учнями необхідних теоретичних і практичних завдань. Ми домовились застосовувати робочі принципи, два з яких (А та Б) уже сформульовані. Але по відношенню до учнів вони мають інтерактивний, тобто спонукальний із зовні, характер.



У той же час будь-який курс повинен *не стільки викладатись, як вивчатись*. Отже необхідна ще хоча б одна робоча ідея, покликана бути *внутрішнім імперативом* або *стимулятором* для учнів. Назвемо її принципом В – **орієнтувати учнів на самостійність у навчанні**.

Один мудрець вказав три шляхи удосконалення: *легкий* – *наслідування*, *важкий* – *досвід* і *шляхетний* – *міркування*. Проаналізуємо процес навчання. У розділі ІІІ вже згадувалась хибна класифікація: навчання буває пасивне, активне та інтерактивне. Ми з цим не погодились. Тому будемо орієнтуватись на таку схему навчання: *учити* – *допомагати* *учитись* – *учитись*. Інтерактивні елементи в навчально-виховному процесі корисно застосовувати не постійно, а епізодично, наприклад, для актуалізації, концентрації, закріплення чи повторення, які в задуманому нами експрес-курсі повинні відігравати допоміжну роль.

Дуже корисно на першому, ознайомлювальному уроці запитати учнів: «Чому ви сьогодні прийшли до кабінету інформатики?». Як правило, останні від несподіванки не знають, що відповісти. Ось зразки відповідей: «вивчати комп'ютери», «згідно розкладу», «грати в комп'ютерні ігри». Але, бачачи,

що вчитель чекає іншої відповіді, учні можуть «гадати» далі: «слухати», «думати», «щоб нас навчили» тощо. Нарешті почується відповідь: «учитись». Це буде тим, чого чекає учитель. Далі слід запитати: «Що означає *учитись*?» Скоріше всього, учні не зможуть правильно відповісти, бо вони ще не розуміють, що *учитись* – це самотійно, без учителя. Тоді їм слід пояснити те, що сказано в двох попередніх абзацах, переконати, що без орієнтації на самонавчання, вони ніколи не знатимуть більше за вчителя, будуть обмежені його знаннями та здібностями. І тільки після цього учні можуть, якщо не сформулювати, то хоча б зрозуміти формулу: ***учитись – це самотійно шукати запитання та відповіді на них.***

Щойно ми продемонстрували ефект несподіванки, висловлений ще одним древнім філософом. Взагалі, в навчально-пізнавальному процесі повинно бути якомога більше несподіваних ситуацій, це приводять до пізнання через здивування. Назвемо його принципом **Г - пізнавати через здивування**, який разом із принципом **В** повинен стати *мотиваційною основою* ЕКАПу. Для забезпечення повноти системи взятих принципів, до них слід долучити ще один – ***принцип творчого співробітництва*** учнів з учителем (Д).

Нижче буде запропоновано орієнтовний ЕКАП з алгоритмізації та програмування мовою Паскаль, адже вище ми домовились розглядати методику вивчення теми «Алгоритмізація і програмування» у такій формі. Кожна с тем, що називаються ***Уроками***, розрахована для вивчення протягом одного уроку, але при обов'язковій умові самотійних відповідей на запитання та виконання завдань, які пропонуються в кінці кожної теми. Курс не слід розглядати, як збірник поурочних планів учителя. Взявши запропонований ЕКАП за зразок, учителі повинні розуміти, що введення в дію принципів А-Д для учнів повинно бути поступовим, обережним, послідовним і не помітним. Наприклад, повністю відмовитись від систематичних повторень вивченого раніше (принцип **Б**) можна лише при забезпеченні свідомого дотримання принципу **А** у поєднанні з мотиваційними принципами (**В-Д**).

ІІІ.3 ВИСНОВКИ ЩОДО ПРИНЦИПІВ ПОБУДОВИ ЕКАП.

Як зазначалось в другому пункті, запропонований ЕКАП повинен відповідати чинній програмі з основ інформатики для універсального профілю [1], тому будуватимемо його синхронно затвердженому МОН України підручнику [15]. Але, дотримуючись вимоги програми щодо кількості відведених навчальних годин і форми експрес-курсу, ми змушені шукати шляхи ущільнення.

У темі «Комп'ютерне моделювання. Основи алгоритмізації та програмування» фактично не відображається *власне програмування*, якщо не брати до уваги кількох окремих згадок терміну «програма», зокрема: «поняття програми», «структура опису алгоритму навчальною алгоритмічною мовою та мовою програмування» і «описання найпростіших алгоритмів навчальною алгоритмічною мовою та мовою програмування».

Підручник [2] містить три розділи: 3. «Етапи розв'язування задач за допомогою комп'ютера», 4. «Вступ до алгоритмізації» та 5. «Початки програмування». Як і частина програми про інформаційну модель, розділ 5 підручника дуже розпливчатий і не конкретний. Там багаторазово у різних контекстах згадується термін «модель», але, на наш погляд, так і нема однозначних відповідей про відмінність понять інформаційної і математичної моделей, комп'ютерного моделювання і побудови моделі, у підручнику немає чіткого і однозначного трактування етапів розв'язування задачі з допомогою комп'ютера. Ми намагатимемось лаконічно і прозоро сформулювати логічний ланцюжок *інформаційна модель – математичне моделювання – етапи комп'ютерного моделювання*

На наш погляд, фактично відсутній чіткий перехід *алгоритм – програма*. Окремий розгляд, як це зроблено в підручнику, вступу до алгоритмізації та початків програмування в умовах обов'язковості практичної форми вивчення теми також не доцільний. Не відходячи від істини, можна вважати, що *програма – це запис алгоритму розв'язання задачі мовою програмування*. Отже є сенс у ЕКАП об'єднати розділи «Вступ до алгоритмізації» та «Початки програмування».

Але для окремого розгляду алгоритмізації та програмування, що традиційно і цілком природньо робиться, є важливі причини. При розгляді питань алгоритмізації *основна увага концентрується на реалізації математичної моделі задачі в алгоритмі*, тобто домінує логічна лінія. Завдяки цьому при вивченні програмування *стає можливим зосередитись виключно на вивченні мови програмування і нюансах запису алгоритму у вигляді програми*. Підручники з мов програмування, як правило дуже об'ємні й детальні, очевидно, що шкільною програмою не передбачено їх вивчення. Отже виникає ще одна проблема: *як в ознайомлювальному курсі засвоїти основи мови програмування?* Ми також не бачимо можливості в нашому експрес-курсі для більш чи менш детального викладання основ мови *Pascal*. Тому нижче будемо розглядати лише вузлові питання запису програм, причому, як уже акцентувалось, по можливості об'єднуючи опис алгоритму і програми. Щоб заповнити прогалини в описі конструкцій *Pascal*, які не важко помітити при дуже стислому викладі матеріалу, ми будемо формулювати питання і ставити завдання, наведені в кінці кожного уроку, так, щоб учень, працюючи з ними, міг самостійно «відкривати» елементарні відомості, опис яких відсутній. Виконуючи завдання, учень буде змушений користуватись електронною довідкою, і таким чином отримувати достатньо повне уявлення про мову програмування. В зв'язку з цим зручно користуватись версією *PascalABC*²⁰.

Піднявши питання перегляду ролі і місця теми «Комп'ютерне моделювання. Основи алгоритмізації та програмування» у курсі сучасної шкільної

²⁰ На рівні програми ЗОШ вона цілком відповідає найчастіше вживаним версіям *Pascal*. Крім того ця версія надає багато переваг, зокрема, допускає «довгі імена», використовує зручнішу систему числових типів, має стандартні підпрограми, наприклад функцію *Power(x,y)*, має русифіковані меню, довідку та ін. Щодо потужності, то практика підтверджує, що ця версія дозволяє реалізувати навіть непересічні «олімпіадні» задачі.

інформатики, ми переслідували ще одну важливу мету. Як уже зазначалось, згідно діючих програм ця тема вивчається фактично оглядово, що затрудняє для учнів її логічне усвідомлення в цілому. Але при цьому, в рекомендованих МОН України підручниках та навчальних посібниках продовжує відображатись класична, обтяжена деталями, схема подання і закріплення навчального матеріалу з алгоритмізації та програмування, яка багато в чому є калькою технологічного, призначеного для майбутніх студентів технічних вузів, підходу до програмування. З позицій шкільного вчителя основ інформатики видно проблеми:

- а) чіткості методологічних та філософських засад вивчення курсу;
- б) підпорядкованості вивчення алгоритмізації та програмування простій і логічній загальній схемі;
- в) слабкості органічного зв'язку із основною частиною курсу основ інформатики.

Вирішення проблеми а) бачиться в усвідомленні та реалізації тези «інформатика іде від математики» та вивченні теми раніше на один рік, тобто в першому семестрі 10-го класу, відразу після ознайомлення з основним системним програмним забезпеченням (операційні системи та сервісне програмне забезпечення) і вивчення простого текстового редактора.

Не виходячи із ліміту часу, передбаченого діючими програмами, подолати проблему б) можна, змінивши хронологію та акценти всередині цієї теми за такою схемою:

1. *Поняття про алгоритми і найпростіші програми.*

- 1.1 Поняття інформаційної та математичної моделі. Етапи розв'язування задач з допомогою комп'ютера.
- 1.2 Алгоритми, їх властивості та виконавці. Система команд виконавця. Форми подання алгоритмів. Поняття програми.
- 1.3 Структура програми. Опис величин мовою програмування. Лінійні програми опрацювання величин. Команди вводу, виводу та присвоєння.
- 1.4 Виконання програми та аналіз її правильності. Покрокове виконання і відлагодження програм.

2. *Структуровані алгоритми та їх програмна реалізація.*

- 2.1 Діалогові програми. Структури повторення та їх графічні схеми.
- 2.2 Структури розгалуження та їх графічні схеми.
- 2.3 Конструювання алгоритмів методами покрокової деталізації. Підпрограми і вказівки їх виклику. Формальні та фактичні параметри.
- 2.4 Структуровані алгоритми.

3. *Структуровані величини та їх опрацювання.*

- 3.1 Структуровані типи величин. Табличні величини (масиви). Лінійні масиви та алгоритми їх опрацювання.
- 3.2 Алгоритми пошуку елементів у таблиці.
- 3.3 Алгоритми упорядкування лінійних таблиць.

3.4 Рядкові величини та алгоритми їх опрацювання.

Згідно цієї схеми тема розбита на три частини по чотири уроки. Перша з них присвячена формуванню понять алгоритма і програми. Як буде видно з додатку 1 «Експрес-курс алгоритмізації та програмування», вже в першій частині наявні дещо відмінні від канонічних трактування алгоритму, команди, програми, процедури, функції, багато уваги приділяється вводу/виводу (з цим пов'язані міркування про лінійні та діалогові алгоритми і програми, в ознайомлювальному плані розглядається поняття текстового файлу та питання використання зовнішніх файлів для вводу/виводу). Це зроблено з тим, щоб учні мали змогу краще уявити структуру і логіку теми та її місце в загальній системі знань.

Акцентовується увага на терміні «*структура*», зокрема, дається чітке поняття *структури алгоритму та програми*. Далі структурованість також є вузловим поняттям, у другій частині ідеться про *структуровані алгоритми*, а в третій про *структуровані типи*.

Ми вважаємо, що навіть при оглядовому вивченні теми повинні детально і чітко розглядатись не тільки питання математичної постановки задачі, а й питання відлагодження та виконання програми, цьому присвячений урок 4.

У другій частині вузловим є питання конструювання програми відповідно принципу покрокової деталізації. Умисно поставлені акценти на конструюванні методом «*згори донизу*», пов'язуючи його з поняттям про стандартні підпрограми та «*знизу догори*», розвиваючи ідею створення процедур і функцій користувача.

Логічним підсумком другої частини є останній урок, де вивченням циклу з параметром закінчується формування уявлень про структуру повторення, що дає змогу розглядати програми, які містять усі базові структури алгоритмів (БСА). Ми вважаємо за доцільне на відміну від того, як це робиться в підручниках, БСА розглядати окремо, вводячи їх поступово, відповідно до ускладнення задач. Це допоможе учням усвідомити суть структурного програмування взагалі.

В третій частині розглядаються найважливіші структуровані типи, зокрема масиви і рядки. У двох перших частинах вони лише згадуються та епізодично використовуються на прикладі типу *string*. Але, доводиться визнати, що, виходячи з ліміту навчального часу, у ЕКАП немає змоги для ознайомлення із записами та множинним типом, а також з рештою порядкових типів, вказівниками та процедурними типами.

Щодо проблеми в) можна сказати наступне. Необхідного органічного зв'язку із основною частиною курсу основ інформатики (інформаційно-комунікаційні технології) можна досягти, по-перше, вирішивши проблеми а)-б), по-друге, забезпечивши наступне після теми «Комп'ютерне моделювання. Основи алгоритмізації та програмування» знайомство із об'єктно орієнтованим програмуванням (ООП) та візуальним проектуванням (ВП) і узгодивши, а в окремих випадках навіть підпорядкувавши, вивчення прикладного програмного забезпечення загального і навчального призначення

(ППЗП, ППЗНП) із темою «Комп'ютерне моделювання. Основи алгоритмізації та програмування».

Кожен урок ЕКАПу є окремою темою, що, здебільшого, не можна вивчити за одну календарну годину, тому, як уже наголошувалось, слід шукати додаткові резерви часу. Але уроки-теми написані так, що на розсуд учителя, їх можна вивчати фрагментарно. Для кращого закріплення у вигляді не обов'язкового додатку в кінці ЕКАПу є практикум із шести практичних робіт, які охоплюють прикладні і практичні задачі.

Поряд із цим відмітимо, що ЕКАП, або його ідея, може успішно застосовуватись і при значно ширшому та глибшому вивченні теми. Його можна розглядати, як *вступ до основ програмування*.

РОЗДІЛ IV.

IV.1 МІЖПРЕДМЕТНІ ЗВ'ЯЗКИ ТА МАТЕМАТИЧНА СКЛАДОВА КУРСУ ОСНОВ ІНФОРМАТИКИ. ЗНАЧЕННЯ МІЖПРЕДМЕТНИХ ЗВ'ЯЗКІВ ДЛЯ ІНФОРМАТИКИ

Однією із пріоритетних у дослідницькій роботі ОШППД була проблема міжпредметних зв'язків інформатики з іншими навчальними предметами. Вони можуть мати різні ступені складності та концентрації, але наявні лише тоді, коли на уроці *акцентується увага на деяких елементах, що належать іншому предмету*. Застосування комп'ютерної презентації на уроці біології або історії зовсім не говорить про міжпредметні зв'язки з інформатикою, тут має місце звичайне застосування засобів однієї галузі для пояснення чи дослідження в іншій, зокрема ІКТ. Міжпредметні зв'язки можуть бути *епізодичними* чи *систематичними* як в межах окремої теми, так і в межах одного уроку. Їх форми достатньо досліджені методистами.

Для інформатики дуже актуальними є міжпредметні зв'язки з фізикою, англійською, українською та російською мовами, причому, але у першу чергу з математикою. Вони завжди дуже різнорівневі та багатогранні, тому підлягають детальному розгляду. При вивченні міжпредметних зв'язків їх слід представляти у вигляді орієнтованого графа і розглядати попарно. Досліджуючи зв'язки між інформатикою та математикою, необхідно вивчати окремо зв'язки *інформатика→математика* та *математика→інформатика*.

Немає сумніву, що математична складова домінує не тільки в темі «Математичне моделювання. Алгоритмізація та програмування», а й у темі «Інформаційно-комп'ютерні технології», тому вона дуже важлива у курсі основ інформатики в цілому. За влучним висловом А.П. Єршова, *інформатика іде від математики*, тому при вивченні інформатики слід використовувати будь-яку, в тому числі не пряму, а опосередковану нагоду підключення математичного компоненту. Навіть розглядаючи перші теми, наприклад, питання про вимірювання інформації, корисно глибоко вивчити позиційні системи числення, акцентувати увагу на питанні про оптимальність розміру базової комірки пам'яті в 1 байт. При цьому слід довести, що 8-бітова комірка може зберігати $2^8=256$ варіантів двійкових кодів і зробити висновок про її оптима-

льність для зберігання інформації. Це приклад *прямого зв'язку математика→інформатика*. Вивчаючи об'єкти у Microsoft Word, корисно звернути увагу на роботу з редактором формул Equation, пропонуючи завдання на створення та редагування математичних виразів. Тут спостерігається *опосередкований зв'язок інформатика→математика*. Прикладів прямого та опосередкованого використання математичного апарату на уроках інформатики можна навести дуже багато.

За відомим висловом М.В. Ломоносова, ***математику ще й тому слід вивчати, що вона розум до ладу приводить***. Це – постулат, який ніким не ставиться під сумнів. Математику, хоча б в обсязі загальноосвітньої школи, вивчають всі люди, бо вона є невід'ємною складовою середньої освіти.

Але важливо постійно думати над розширенням традиційних меж використання елементів математики при вивченні інформатики. Зауважимо, що цілі математичні теорії проникли у всі галузі людської діяльності як на науковому, так і на побутовому рівні. У першу чергу це стосується арифметики та логіки. Наприклад, у побуті часто користуються логічним апаратом, не думаючи, що тут має місце пряме застосування математики. На наш погляд, інформатика, як прикладна галузь знань, *дає своєрідне уточнення та узагальнення не тільки таких понять, як інформація та алгоритм, а й багатьох загальноосвітніх термінів, від лінгвістичних, наприклад, «алфавіт» та «мова», до природничих і математичних, наприклад, «число», «величина», «тип», або «вид», «вираз», «формула», «функція», «процедура» тощо*. Іншими словами, ***інформатика, покликана робити наступний за математикою крок у «приведенні розуму до ладу»***.

Про що тут ідеться? Візьмемо поняття «алфавіт». У початкових класах його формують у вузькому розумінні, лише як множину знаків, що позначають звуки конкретної національної мови. Але чи правильно, коли дорослі люди так розуміють поняття «алфавіт» до кінця життя? Навіть у XXI столітті ми спостерігаємо, до яких тяжких для людства наслідків приводить довільне трактування історичних, політичних, економічних і взагалі – наукових понять. Тому без перебільшення можна стверджувати, що все починається з дуже приблизного уявлення про алфавіт. На уроках інформатики при вивченні мови програмування учні дізнаються, що алфавіт у *широкому розумінні* – це набір *усіх* символів, необхідних для висловлення думок. А потім переконуються, що, до українського алфавіту на рівних правах, наприклад, з буквою «а», входять арабські цифри, розділові знаки і т.д. Мовознавці мабуть сперечались би з цього приводу. Але ж не заперечать вони, що величезна кількість слів іншомовного походження, зокрема, слово «комп'ютер» – не належать до українського словника. Якщо формування такого уявлення про алфавіт не передбачено в початкових класах та на уроках рідної мови, то вивчення мови програмування доцільне хоча б з огляду на це. Але, на наше глибоке переконання, будь-який учитель-класовод чи мовник повинен у свій час узагальнити учням поняття алфавіту, бо неможливе розуміння процесу

пізнання без принципу «від споглядання до узагальнення», адже аналіз та синтез – це альфа та омега мислительського процесу.

Порівняймо поняття величини чи функції у математиці та фізиці – з одного боку, та в інформатиці – з іншого. Величина у першому випадку має ха-рактеристики: *ім'я* і *значення*. Про тип величини майже не згадується, адже з точки зору математики і фізики величини фактично бувають лише числові, правда існує поділ на скалярні та векторні. З програмування учні дізнаються про кодування величин, про потребу їх оголошення та опису, про різноманіття типів, зокрема у мові Паскаль, у якій існує по кілька цілочисельних та дробових типів. На перший погляд це дивує, але коли пояснити про представлення величини у пам'яті комп'ютера, здивування проходить. Ім'я величини в математиці не обговорюється. Час – t , швидкість – v і т.д. Мови про позначення величини, як деякого набору допустимих символів, нема. Але про це доводиться говорити в інформатиці, де ім'я, чи ідентифікатор може бути не тільки у величини, а й у файлу, папки, диску, програми тощо. Коли пояснюєш, що ім'я, в тому числі і людини – це лише позначення, деякий набір символів, у багатьох виникає протиріччя, пов'язане із засвоєною раніше абсурдною і шкідливою для наукового світогляду містикою імен. Але учні перестають дивуватись і погоджуються, коли їм говорять, що ім'я «Володимир» для американського індіанця так само є лише набором звуків, як для нас «Чингачгук». Таким чином реалізується *процес пізнання через здивування*. Лише сміх замість здивування викликало повідомлення, що кандидатом у президенти в жовтні 2009 р. став громадянин В. Гуменюк, який для реєстрації змінив своє прізвище на «Противсіх»). Отже, щоб не виникало дилетантських уявлень, необхідно ставити наукові (читай логічні, математичні) акценти.

Необхідно хоча б коротко пригадати поняття функції. Після досить детального ознайомлення із ним у школі, багатьом студентам на лекціях з математичного аналізу лише завдяки великим зусиллям вдається зрозуміти класичні означення функції. Але, щоб досягти глибини усвідомлення, у математиці уявлення про функцію розглядається дуже вузько. В інформатиці необхідно акцентувати, що функція - це не лише залежність між двома змінними, при якій кожному значенню однієї відповідає єдине значення другої, а й *дія*, яка певне значення аргументу перетворює у цілком визначене значення функції. Для учнів цікаво, що термін «дія» тут слід розуміти, як «автомат», подібно тому, як електром'ясорубка – це автомат для перетворення м'яса у фарш. У процесі таких міркувань свідомість учнів робить колосальний світоглядний скачок, багато різних за формою явищ зводяться, так би мовити, «до спільного знаменника». А якщо пригадати, що математика, навіть вища, практично обмежується розглядом функцій двох числових аргументів, а інформатика, точніше розділ «Алгоритмізація і програмування», розглядає поняття функції багатьох аргументів, причому, не тільки числових типів, а саме поняття функції у певному розумінні зводиться до поняття *процедури*, то можна сказати, що абстрактне математичне поняття функції в інформатиці набуває живий, зрозумілий без складного математичного апарату образ. От і

спробуй не погодитись, що інформатика, як сказано дуже доречно вище, доповнює математику у «приведенні розуму до ладу», тобто у формуванні наукового світогляду.

Є цілий ряд тем, у яких математичний компонент домінуючий. Класичним прикладом є тема «електронні таблиці», адже основне завдання табличного процесора *Microsoft Excel* - це автоматизація обчислень. Але і тут, якщо не слідувати сліпо за підручником [15], є невичерпні додаткові можливості непрямого використання математики. Треба лише творчо підходити до вибору завдань. Для опису формування *робочого листа таблиці Microsoft Excel*, дій з *майстрами і шаблонами* у підручнику розглянуто, задачі на рух та про визначення кількості мікробів у класі, що припадають на одного учня у різний час і при різній температурі повітря. Ефективнішими при вивченні формул та майстра функцій були б приклади створення таблиці, що містить, наприклад, довільну кількість квадратних рівнянь і їх розв'язків, чи математичні таблиці Брадіса для визначення значень тригонометричних функцій (синус-косинус, тангенс-котангенс) для кута α ($0^\circ < \alpha < 90^\circ$), причому не тільки чотиризначних, а й для більшої розрядності. Дуже цінним математичним моментом при виконанні цього завдання є обчислення та відображення кутів у градусній мірі. Взагалі ж цінність міжпредметних зв'язків інформатики з математикою у можливостях проведення чисельних експериментів, досліджень, аналізу та синтезу, побудови математичних моделей, як пропедевтичний етап для програмування.

Сказати, що таких зразків прямого й доцільного використання математики при вивченні будь-яких розділів інформатики можна навести дуже багато, було б неправильним. Насправді, в курсі інформатики, «куди не кинь», там і наткнешся на математику. Требі тільки це помічати і вміти використовувати. Так само широко на уроках математики можна включати можливості ІКТ та відомості з основ інформатики. Ось ще приклади. Приблизно одночасно у восьмих класах на уроках алгебри вивчають арифметичний квадратний корінь, ірраціональні та дійсні числа, у геометрії задачі на побудови вписаних чи описаних трикутників та чотирикутників, а на уроках інформатики побудову графіків та діаграм. Все це можна ефективно поєднати. Демонструючи побудову у *Microsoft Word* малюнків, графіків функцій, зразки розв'язань варіантів контрольної роботи з геометрії чи розбір завдань олімпіади з математики, можна піднести інтерес до програмного забезпечення та вивчення основ інформатики, показати якісно відмінну інформатику від «зависання» в Інтернеті або від «різання» до виснаження у комп'ютерні ігри.

IV.2 БІНАРНІ УРОКИ.

Отже, можна вести мову про *уроки інформатики із математичними елементами* та *уроки математики із застосуванням інформатики*. Найвищий ступінь міжпредметних зв'язків досягається в умовах проведення *бінарних уроків*. При вивченні інформатики бінарні уроки заслуговують на особливу увагу, тому їм майже цілком було присвячено восьме заняття ОШПІД.

Але такий тип уроків слід відрізняти від звичайних *міжпредметних* та *інтегрованих* уроків.

Міжпредметний урок – це форма занять, яка застосовується на всіх ступенях навчання. Його особливості – *навчальний матеріал ілюструється відомостями з інших предметів*, забезпечує при цьому синхронність навчання по темах кількох предметів, що перетинаються, які розділені в часі на місяці або роки. Міжпредметний урок може вести один учитель.

Інтегрований урок – це заняття, на якому *визначена тема розглядається з різних точок зору*, засобами деяких предметів чи курсів. Ведуть його кілька учителів. Наприклад: пресс-конференція, конференція-захист, круглий стіл тощо.

У методичній літературі *бінарні уроки* та їх завдання трактуються по різному. Переважає застаріле, надто спрощене уявлення, коли під бінарними маються на увазі всі уроки із елементами міжпредметних зв'язків, уроки з використанням інформаційно-комп'ютерних технологій, навчаючих програм тощо.

Інший погляд на бінарні уроки трактує їх, як варіанти *нетрадиційних уроків*²¹, що часто використовуються в інноваційній роботі та спрямовані на *об'єднання окремих тем суміжних предметів у рамках одного уроку*, наприклад фізики та біології, математики та фізики, математики та хімії, хімії та біології. Інакше кажучи, бінарним уроком вважається навчальне заняття, яке об'єднує вивчення тем двох предметів одного циклу (або освітньої галузі) в одному уроці. З таким трактуванням слід погодитись, але із деякими застереженнями. Адже дві різні теми не можна вивчати *одночасно і паралельно*. Їх можна *об'єднати в одну*, ширшу тему, або вивчати послідовно, *застосовувавши результати вивчення першої для вивчення другої*. Тобто, у бінарних уроках *виклад чи дослідження проблеми одного предмету знаходить продовження в іншому*, міжпредметні зв'язки реалізуються в процесі викладання дисциплін однієї освітньої галузі.

Традиційно вважається, що бінарний урок як правило проводять два учителі, а тому *основною метою бінарних уроків вважається розвиток співробітництва педагогів і формування в учнів переконання в зв'язності та єдності предметів*, що не може становити особливої цінності для процесу навчання. Дійсно, чи має *принципове значення* для вивчення окремих предметів *співробітництво двох педагогів*? Відчуття зв'язності та єдності предметів також не може бути самоціллю, адже воно природньо виникає на етапі узагальнення і є функцією міцності засвоєння знань, умінь та навичок.

Дотримуючись вищеописаних ознак та вимог, інколи можна «створити» (що й зустрічається в методичній літературі) досить незвичайний, зовні, а по змісту навіть казуїстичний зразок уроку: учитель математики проводить зі своїм колегою по фізкультурі бінарний урок, наприклад, «Естафета с математичними завданнями». Пропонується перекинутись уперед через голову, взяти картку, пробігти по перевернутій лаві, зупинитись, визначити, чи стандартний вигляд одночлена записано на картці, і покласти картку на підлогу.

²¹ Бінарний урок за Ю.А. Самарінім).

Наступні вправи естафети: добігти до парти, записати одночлен в стандартному вигляді, назад рухатись стрибками на правій нозі по лаві, перекинутись уперед через голову.

Згідно опису Ю.А. Самаріна даний вид заняття цікавий тим, що можна активізувати мислительську енергію через фізичні вправи. Заняття проходять на музичному фоні. Учні одержують по декілька оцінок за урок.

Правда там же можна зустріти застереження, що такі бінарні уроки не можуть застосовуватись часто у педагогічній практиці через їхню складність у підготовці, а також через протипоказання: в застосуванні бінарних уроків є неузгодженість та несумісність педагогів (протипоказань можна знайти ще).

Виходячи з вищесказаного, на нашу думку, можна так сформулювати поняття бінарного уроку: *Бінарними* слід вважати лише такі уроки з міжпредметними зв'язками із *двох предметів*, коли питання, що розглядаються в темі уроку, *стосуються обох предметів* і *між собою логічно поєднані*, тобто, **бінарні уроки поєднують два предмети в одній темі** [26].

Наприклад, можна говорити про бінарні уроки *інформатика – математика*. Безперечно, на кожному бінарному уроці повинна бути встановлена пріоритетність одного із предметів, іншими словами, можуть бути бінарні уроки інформатика–математика і математика–інформатика. Компоненти уроку, який має вищий пріоритет назвемо *домінантними*, а іншого – *підпорядкованими*. Але *урок*, в тому числі й бінарний, *повинен проводити*, як правило, *один учитель*.

Зразок бінарного уроку з математики та інформатики описаний у [19]. Але там, як указано в темі, ідеться про урок геометрії «Паралельність прямих і площин у просторі» з використанням навчальної програми «Репетитор по математике Кирилла и Мефодия» та графічного редактора *Paint*, тобто у темі уроку не передбачено розгляд питань з інформатики, а питання про графічні системи, формати графічних файлів, формування кольорів, які в ході уроку все ж ставляться, та самостійна робота за варіантами на побудову стереометричних фігур з допомогою *Paint*, не пов'язані з темою «Паралельність прямих і площин у просторі». Тому тут скоріше ідеться про урок з використання ІКТ, ніж про бінарний урок.

IV.3 ЗРАЗКИ БІНАРНИХ УРОКІВ.

Наведемо приклади бінарних уроків *математика–інформатика* та *інформатика–математика*, які відповідають сформульованому вище уявленню про них.

*бінарний урок²²: **математика – інформатика**²³,
проводиться в кабінеті ОІОТ або в кабінеті ІКТН*

тема: ТРИГОНОМЕТРИЧНІ ТАБЛИЦІ ДЛЯ ОБЧИСЛЕННЯ ЗНАЧЕНЬ СИНУСА, КОСИНУСА, ТАНГЕНСА І КОТАНГЕНСА ГОСТРИХ КУТІВ.

²² Урок для 8 класу, в якому вивчається інформатика відповідно [17].

²³ Домінантні елементи (математика) виділено суцільною рамкою, підпорядковані елементи (основи інформатики) виділено пунктирною рамкою.

мета: Розглянути створення і застосування тригонометричних таблиць кутового аргумента.

обладнання та література:

- М.І. Бурда, Л.М. Савченко, Геометрія, навчальний посібник для 8-9 класів шкіл з поглибленим вивченням математики, Київ, «Освіта», 2004 р.;
- І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов, Інформатика, частина II, підручник для 10-11 класу загальноосвітніх навчальних закладів, Київ, «Форум», 2004 р.;
- *Microsoft Office Excel 2003*.
- П'ятизначні тригонометричні таблиці гострих кутів (додатки 1-2).
- Таблиця *Sin-Cos-Tg-Ctg.cls*

ПЛАН УРОКУ

1. Повторення

а) Повторити поняття синуса, косинуса, тангенса і котангенса гострих кутів числового та кутового аргументів.

б) Повторити таблицю значень синуса, косинуса, тангенса та котангенса 30° , 45° , 60° :

	<i>sin</i>	<i>cos</i>	<i>tg</i>	<i>ctg</i>
30°	$\frac{1}{2} \approx 0,5$	$\frac{\sqrt{3}}{2} \approx 0,8660$	$\frac{\sqrt{3}}{3} \approx 0,5774$	$\sqrt{3} \approx 1,7321$
45°	$\frac{\sqrt{2}}{2} \approx 0,7071$	$\frac{\sqrt{2}}{2} \approx 0,7071$	1	1
60°	$\frac{\sqrt{3}}{2} \approx 0,8660$	$\frac{1}{2} \approx 0,5$	$\sqrt{3} \approx 1,7321$	$\frac{\sqrt{3}}{2} \approx 0,5774$

в) Повторити формули переведення кутів з градусної міри в радіанну і навпаки:

$$360^\circ = 2\pi \text{ рад} \Rightarrow \begin{cases} 1^\circ = \frac{\pi}{180} \text{ рад} \Rightarrow n^\circ = \frac{n \cdot \pi}{180} \text{ рад}; \\ 1 \text{ рад} = \left(\frac{180}{\pi}\right)^\circ \Rightarrow k \text{ рад} = k \cdot \left(\frac{180}{\pi}\right)^\circ \end{cases} \quad (1)$$

г) Повторити формули переведення кутів з *градусів* у *хвилини* і навпаки:

$$1^\circ = 60' \Rightarrow \begin{cases} 1' = \left(\frac{1}{60}\right)^\circ \Rightarrow n' = \left(\frac{n}{60}\right)^\circ; \\ k^\circ = 60k' \end{cases} \quad (2)$$

д) Повторити вставку формул у комірки таблиць *Excel*.

Звернути увагу на те, що формула записується у *рядку формул* і починається значком «=» (чому?!), а після її занесення у *комірку* відображається значення, обчислене за формулою.

	A	B	C
1	1	градуси	0.15
2	2	3	0.2
3	3	12	0.216667
4	4	14	0.233333
5	5	16	0.25

2. Вивчення нового матеріалу

I. Постановка та розв'язання проблем:

Розглянути на прикладах застосування тригонометричних таблиць для гострих кутів:

а) таблиці у текстовому форматі: *Sin-Cos; Tg-Ctg* – додатки 2-3

– Як обчислити значення *sin, cos, tg, ctg* для кутів:

1) 12° ; 2) $12^\circ 6'$?

– Чи можна обчислити значення *sin, cos, tg, ctg* для кутів:

3) $12^\circ 7'$; 4) $12^\circ 10'$ з допомогою даних друкованих таблиць? Чому?

б) Зміни в таблицях текстового формату.

Слід звернути увагу на те, що подібні традиційні друковані таблиці, як показано нижче, (див. таблицю 1) мають поправки на $1'$, $2'$, та $3'$.

таблиця 1

СИНУСИ											поправки:			
	0	6	12	18	24	30	36	42	48	54		1'	2'	3'
0	0	0,00175	0,00349	0,00524	0,00698	0,00873	0,01047	0,01222	0,01396	0,01571	89	0,00058	0,00104	0,00174

Проте практично такої потреби немає, адже ці поправки можна легко розрахувати вручну, знайшовши різницю двох сусідніх значень і поділивши її на 6.

Тригонометричні таблиці, наведені в додатках 2-3, можна легко створити з допомогою *Excel* (таблиця 2).

в) таблиці у форматі *Excel*: *Sin-Cos-Tg-Ctg – Excel*

таблиця 2

D4							D5						
=SIN((A4+12/60)*ПИ()/180)							=SIN((A5+D2/60)*ПИ()/180)						
A	B	C	D	E	F		A	B	C	D	E	F	
1						СИН	1						СИН
2		0	6	12	18	24	2		0	6	7	18	24
3	0	0	0,00175	0,00349	0,00524	0,00698	3	0	0	0,00175	0,00204	0,00524	0,00698
4	1	0,01763	0,0192	0,02094	0,02269	0,02443	4	1	0,01763	0,0192	0,01949	0,02269	0,02443
							5	2	0,0349	0,03693	0,04013	0,04188	

Як видно з таблиці 2, в рядку формул застосовано формули (1) та (2), що розглядаються при повторенні²⁴.

II. Зміни в таблицях формату *Excel*.

В електронних тригонометричних таблицях, створених з допомогою *Excel*, у рядку формул можна змінити значення, що додається до аргументу в стовпчику А, (порівняйте таблиці 2-3), наприклад, в таблиці 3 значення комірки B_3 дорівнює $2'$, відповідно B_5 дорівнює $60'$.

таблиця 3

C5							=SIN((A5+B5/60)*ПИ()/180)						
A	B	C	D	E	F		A	B	C	D	E	F	
1						Синус, косинус, тангенс і котангенс гострого кута	1						
2	градуси	мінути	синус	косинус	тангенс	котангенс	2	градуси	мінути	синус	косинус	тангенс	котангенс
3	3	2	0,052917	0,998599	0,052991	18,87107	3						
4	30	0	0,5	0,866025	0,57735	1,732051	4						
5	59	60	0,866025	0,5	1,732051	0,57735	5						

3. Закріплення

а) Користуючись таблицями (додатки 2-3), визначити $\sin 2^\circ 60'$; $\sin 3^\circ 0'$. Результати порівняти;

б) З допомогою *Excel* модернізувати таблицю 2, яка містить *sin-cos* кутів, значення яких відрізняються на $3'$;

в) Модернізувати таблицю 2 так, щоб вона містила значення *sin-cos* з указаною кількістю знаків після коми.

²⁴ Тут реалізується основний принцип, що лежить в основі сформульованого вище уявлення про бінарний урок.

з) Перевірити правильність формул

для значень α : 1) 12° ; 2) $12^\circ 6'$; 3) $12^\circ 7'$; 4) $12^\circ 10'$.

4. Висновки та підбиття підсумків

Слід звернути увагу на складність і громіздкість процесу створення тригонометричних таблиць без використання Excel, на можливості інтенсифікації обчислювальних процесів з допомогою комп'ютерів. Урок має комбінований характер, але бажано не упустити жодної можливості оцінки роботи учнів як з математики, так і з інформатики.

5. Домашнє завдання²⁵

додаток 1

П'ЯТИЗНАЧНІ ТРИГОНОМЕТРИЧНІ ТАБЛИЦІ СИНУСІВ І КОСИНУСІВ (фрагментарно)

СИНУСИ											
	0'	6'	12'	18'	24'	30'	36'	42'	48'	54'	
0°	0	0,00175	0,00349	0,00524	0,00698	0,00873	0,01047	0,01222	0,01396	0,01571	89°
1°	0,01745	0,0192	0,02094	0,02269	0,02443	0,02618	0,02792	0,02967	0,03141	0,03316	88°
2°	0,0349	0,03664	0,03839	0,04013	0,04188	0,04362	0,04536	0,04711	0,04885	0,05059	87°
3°	0,05234	0,05408	0,05582	0,05756	0,05931	0,06105	0,06279	0,06453	0,06627	0,06802	86°
4°	0,06976	0,0715	0,07324	0,07498	0,07672	0,07846	0,0802	0,08194	0,08368	0,08542	85°
5°	0,08716	0,08889	0,09063	0,09237	0,09411	0,09585	0,09758	0,09932	0,10106	0,10279	84°
6°	0,10453	0,10626	0,108	0,10973	0,11147	0,1132	0,11494	0,11667	0,1184	0,12014	83°
7°	0,12187	0,1236	0,12533	0,12706	0,1288	0,13053	0,13226	0,13399	0,13572	0,13744	82°
8°	0,13917	0,1409	0,14263	0,14436	0,14608	0,14781	0,14954	0,15126	0,15299	0,15471	81°
9°	0,15643	0,15816	0,15988	0,1616	0,16333	0,16505	0,16677	0,16849	0,17021	0,17193	80°
10°	0,17365	0,17537	0,17708	0,1788	0,18052	0,18224	0,18395	0,18567	0,18738	0,1891	79°
81°	0,98769	0,98796	0,98823	0,98849	0,98876	0,98902	0,98927	0,98953	0,98978	0,99002	8°
82°	0,99027	0,99051	0,99075	0,99098	0,99122	0,99144	0,99167	0,99189	0,99211	0,99233	7°
83°	0,99255	0,99276	0,99297	0,99317	0,99337	0,99357	0,99377	0,99396	0,99415	0,99434	6°
84°	0,99452	0,9947	0,99488	0,99506	0,99523	0,9954	0,99556	0,99572	0,99588	0,99604	5°
85°	0,99619	0,99635	0,99649	0,99664	0,99678	0,99692	0,99705	0,99719	0,99731	0,99744	4°
86°	0,99756	0,99768	0,9978	0,99792	0,99803	0,99813	0,99824	0,99834	0,99844	0,99854	3°
88°	0,99939	0,99945	0,99951	0,99956	0,99961	0,99966	0,9997	0,99974	0,99978	0,99982	1°
89°	0,99985	0,99988	0,9999	0,99993	0,99995	0,99996	0,99998	0,99999	0,99999	1	0°
	54'	48'	42'	36'	30'	24'	18'	12'	6'	0'	
КОСИНУСИ											

додаток 2

П'ЯТИЗНАЧНІ ТРИГОНОМЕТРИЧНІ ТАБЛИЦІ ТАНГЕНСІВ І КОТАНГЕНСІВ (фрагментарно)

ТАНГЕНСИ											
	0'	6'	12'	18'	24'	30'	36'	42'	48'	54'	
0°	0	0,00175	0,00349	0,00524	0,00698	0,00873	0,01047	0,01222	0,01396	0,01571	89°
1°	0,01746	0,0192	0,02095	0,02269	0,02444	0,02619	0,02793	0,02968	0,03143	0,03317	88°
2°	0,03492	0,03667	0,03842	0,04016	0,04191	0,04366	0,04541	0,04716	0,04891	0,05066	87°
3°	0,05241	0,05416	0,05591	0,05766	0,05941	0,06116	0,06291	0,06467	0,06642	0,06817	86°

²⁵ Конкретно можна визначити лише врахувавши обставини, у яких проведено урок, тому ми віднесемо це до компетенції учителя, що проводив урок.

4°	0,06993	0,07168	0,07344	0,07519	0,07695	0,0787	0,08046	0,08221	0,08397	0,08573	85°
5°	0,08749	0,08925	0,09101	0,09277	0,09453	0,09629	0,09805	0,09981	0,10158	0,10334	84°
6°	0,1051	0,10687	0,10863	0,1104	0,11217	0,11394	0,1157	0,11747	0,11924	0,12101	83°
7°	0,12278	0,12456	0,12633	0,1281	0,12988	0,13165	0,13343	0,13521	0,13698	0,13876	82°
8°	0,14054	0,14232	0,1441	0,14588	0,14767	0,14945	0,15124	0,15302	0,15481	0,1566	81°
9°	0,15838	0,16017	0,16196	0,16376	0,16555	0,16734	0,16914	0,17093	0,17273	0,17453	80°
10°	0,17633	0,17813	0,17993	0,18173	0,18353	0,18534	0,18714	0,18895	0,19076	0,19257	79°

81°	6,31375	6,38587	6,45961	6,53503	6,61219	6,69116	6,77199	6,85475	6,93952	7,02637	8°
82°	7,11537	7,20661	7,30018	7,39616	7,49465	7,59575	7,69957	7,80622	7,91582	8,02848	7°
83°	8,14435	8,26355	8,38625	8,51259	8,64275	8,77689	8,9152	9,05789	9,20516	9,35724	6°
84°	9,51436	9,6768	9,84482	10,0187	10,1988	10,3854	10,5789	10,7797	10,9882	11,2048	5°
85°	11,4301	11,6645	11,9087	12,1632	12,4288	12,7062	12,9962	13,2996	13,6174	13,9507	4°
86°	14,3007	14,6685	15,0557	15,4638	15,8945	16,3499	16,8319	17,3432	17,8863	18,4645	3°
88°	28,6363	30,1446	31,8205	33,6935	35,8006	38,1885	40,9174	44,0661	47,7395	52,0807	1°
89°	57,29	63,6567	71,6151	81,847	95,4895	114,589	143,237	190,984	286,478	572,957	0°
	54'	48'	42'	36'	30'	24'	18'	12'	6'	0'	
КОТАНГЕНСИ											

бінарний урок: *інформатика–математика*²⁶,
проводиться в кабінеті ОІОТ

тема: **МАТЕМАТИЧНА ПОСТАНОВКА ЗАДАЧІ. ЕТАПИ РОЗВ'ЯЗУВАННЯ ЗАДАЧ З ДОПОМОГОЮ КОМП'ЮТЕРА.**

мета: Розглянути питання про моделювання; Вивчити поняття інформаційної та математичної моделі; Сформувати навички математичної постановки задачі.

обладнання та література:

- І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов, Інформатика, частина II, підручник для 10-11 класу загальноосвітніх навчальних закладів, Київ, "Форум", 2004 р.;
- Microsoft Office Word 2003, Microsoft Office Excel 2003.
- Таблиця "Характеристики величин"

ПЛАН УРОКУ

1. Повторення та актуалізація

а) Розв'язати у зошиті поваріантно задачі:

I варіант

Натрапивши на капустяне поле, що складається з 40 рядків, у кожному з яких росте по 200 капустин, Заєць вирішив влаштувати бенкет для своїх Вухатих друзів. Скільки Вухатих друзів зможе запросити Заєць на бенкет, якщо за ніч вони з'їдають по 16 качанів капусти?

Розв'язання:

$$40 * 200 / 16 = 500.$$

Відповідь: 500 Вухатих друзів

II варіант.

За даними довжиною і шириною прямокутника визначити його площу. Обчислити, якщо довжина дорівнює 40 см, ширина дорівнює 25 см.

Розв'язання:

Позначимо довжину прямокутника a , ширину b , площу S .
 $S = a * b$, при $a = 40$, $b = 25$
 $S = 40 * 25 = 1000 \text{ (см}^2\text{)}.$

Відповідь: 1000 см^2

²⁶ Домінантні елементи (інформатика) виділено суцільною рамкою, підпорядковані елементи (математика) виділено пунктирною рамкою.

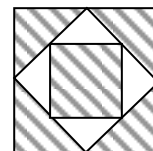
Висновок: Задачі ідентичні по змісту, бо обидві розв'язуються за формулою $S=a*b$, але друга додатково потребує позначення даних (аргументів) та результату. Узагальнення умови подібних задач, необхідне для їх розв'язування, називають *математичною постановкою*, яка складається з двох етапів:

- 1) опис параметрів (аргументів та результатів);
- 2) побудова математичної моделі.

б) За даним зразком у текстовому документі *Word* побудувати поваріантно малюнки при даних n – кількість сторін правильного многокутника та k - кількість вписаних многокутників:

I варіант: $n = 5$; $k = 3$.

II варіант: $n = 4$; $k = 5$.



в) Користуючись *Microsoft Excel*, самостійно заповнити таблицю:

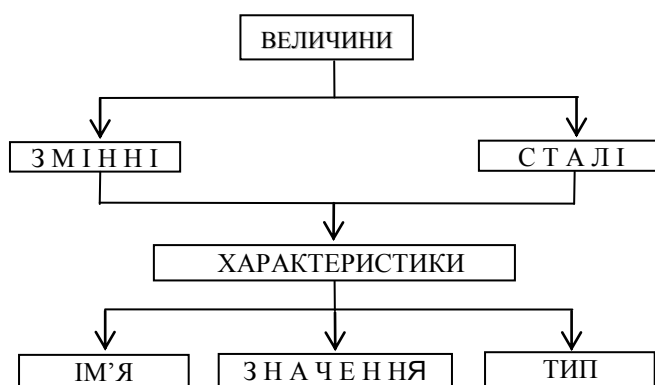
a	b	S
2	12	
4	3	
6	-6	
8	-15	
10	-24	
12	-33	

2. Вивчення нового матеріалу

Під *моделлю* прийнято розуміти *точну або подібну копію деякого об'єкта*, яку можна взяти за зразок, наприклад, модель (марку) мобільного телефона, зменшену, але діючу копію літака тощо.

Інформаційною моделлю будемо вважати *точний опис* з допомогою тексту або аудіо-візуальних засобів *деякого матеріального чи абстрактного об'єкта або явища*, тобто *повну інформацію* про нього.

Кожна модель має *параметри*, тобто *величини*, що виражають її форму, розміри і т.д. Величини діляться на постійні (константи) та змінні і мають три характеристики: *ім'я* (інакше - позначення або ідентифікатор), *тип* та *значення*.



Математичною моделлю називається *опис деякого об'єкта чи явища засобами математики, тобто у вигляді формул*.

Але для запису формул необхідно встановити, тобто описати величини, які будуть даними (*аргументами*), допоміжними та шуканими (*результатами*). Як було сказано в п.1, опис величин та побудова математичної моделі називається *математичною постановкою задачі*. Математична постановка – це перший етап розв'язування задачі, що передуює визначенню її результатів. На уроках математики, фізики та інших предметів визначення результатів виконують особисто, тобто вручну, інколи для проміжних обчислень

використовуючи калькулятор чи якусь спеціальну комп'ютерну програму, наприклад *Microsoft Excel*.

Існують задачі, що передбачають автоматизацію вводу аргументів, обчислень та виводу результатів через їх дуже велику кількість, тому розв'язуються з допомогою комп'ютера в автоматичному режимі. Процес розв'язування таких задач складається з кількох етапів, зокрема:

- a) Виконання математичної постановки задачі;
- b) Опису, дослідження та доведення алгоритму;
- c) Складання та тестування (перевірка правильності) програми;
- d) Виконання програми і одержання результатів (відпові задачі).

Після математичної постановки задачі у «ручному» режимі виконуються необхідні обчислення і одержуються результати. При розв'язуванні задачі з допомогою комп'ютера в автоматичному режимі потреби обчислювати особисто немає, замість цього слід виконати п. b) - d).

3. Закріплення

Приклад 1 (Сума) Обчислити суму перших 2500 натуральних чисел. Виконати математичну постановку розв'язання цієї задачі для n перших натуральних чисел.

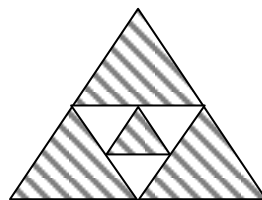
Уявивши вираз для обчислення суми перших 2500 натуральних чисел: $1+2+3+4+5+\dots+99+100+101+\dots+1000+1001+\dots+2499+2500$, та час, потрібний для обчислення вручну, доведеться погодитись, що розв'язування цієї задачі слід почати з другої частини, тобто з математичної постановки.

Математична постановка:

- **Опис величин.** Нехай n – кількість доданків, а S_n – їх сума. Обидві величини повинні бути натуральними числами.
- **Математична модель.** Записавши шукану суму та підмітивши, що суми доданків, однаково віддалених від кінців, рівні між собою, а їх кількість у двічі менша кількості доданків, одержимо: $S_{2500} = (1+2500) + (2+2499) + \dots + (1250+1251)$, після узагальнення, одержимо математичну модель: $S_n = (1+n) \cdot n/2$.

Обчислення. При $n = 2500$ $S_n = (1+2500) \cdot 2500/2 = 3126250$.

Приклад 2 (Фігура) Виконати математичну постановку задачі обчислення площі заштрихованої фігури, що складається з n вписаних правильних трикутників, якщо сторона зовнішнього трикутника a і заштриховано трикутники з непарними номерами (на малюнку 1 $n = 3$)



Математична постановка:

- **Опис величин.** Нехай n – кількість доданків, a – сторона зовнішнього правильного трикутника, а s_n – площа заштрихованої фігури. Додатково позначимо площу зовнішнього трикутника s_0 . На відміну від попередніх прикладів, якщо n – натуральне, то a , s_n та s_0 – повинні бути дійсними числами.
- **Математична модель.** Легко помітити, що площа вписаного трикутника завжди в четверо менша за площу описаного трикутника, тому: $s_n = s_0 - s_0/4 + s_0/16 - s_0/64 + s_0/256 - \dots$. Але ця формула з двома суттєвими недоліками: у виразі відсутній параметр n , вона має два варіанти – для парних і непарних n , крім того, s_0 слід виразити через a , тобто її слід узагальнити. Для цього достатньо помітити, що доданки становлять геометричну прогресію з першим членом s_0 та знаменником, рівним $-1/4$. Отже:

$$s_n = \frac{s_0 \left(\left(-\frac{1}{4} \right)^n - 1 \right)}{-\frac{1}{4} - 1}, \text{ при } s_0 = \frac{a^2 \sqrt{3}}{4}.$$

Тому остаточний вигляд формули такий: $s_n = \frac{a^2 \sqrt{3} \left(\left(-\frac{1}{4} \right)^n - 1 \right)}{5}.$

Наведені формули – це два варіанти математичної моделі даної задачі. Але завжди слід визначати остаточний, найбільш загальний варіант, тому математичною моделлю задачі будемо другу формулу.

I. Усні питання:

- I.1 Чи можна вважати інформаційну модель математичною моделлю?
- I.2 Чи можна вважати математичну модель інформаційною моделлю?
- I.3 Яка різниця між інформаційною та математичною моделями?
- I.4 Чи є в прикладах II.1-II.3 допоміжні величини, тобто ті, які не можна назвати ні аргументами, ні результатами?
- I.5 Чи можна на власний розсуд вказувати типи величин, виконуючи математичну постановку задачі?
- I.6 Чи можуть аргументи бути константами? змінними?
- I.7 Чи можуть результати бути константами? змінними?
- I.8 Чи можуть допоміжні величини бути константами? змінними?

- II. *Самостійне завдання (у зошитах):* За даним радіусом кола визначити його довжину та площу відповідного круга. Виконати математичну постановку та для даного значення аргумента обчислити результати.

4. Домашнє завдання

- 1) Повторити послідовності, арифметичну та геометричну прогресії;
- 2) Вивчити поняття математичної моделі, математичну постановку задачі;
- 3) Виконати завдання:

Завдання 1 Обчислити кількість усіх діагоналей для опуклого n -кутника при даному n . Виконати математичну постановку розв'язання задачі.

Завдання 2 Виконати математичну постановку задачі обчислення площі заштрихованої фігури, що складається з n вписаних квадратів, якщо сторона зовнішнього квадрата a і заштриховано квадрати з непарними номерами (на малюнку 2 $n = 3$)

IV.4 ЗВ'ЯЗКИ ІНФОРМАТИКИ З ВИХОВНОЮ РОБОТОЮ.

Сьогодні часто говорять про проекту роботу в школах, але рідко, та й то лише у формальному виступі на конференції, секції чи педраді можна почути про творчі ідеї, над якими працює конкретний педагогічний колектив. Колись колективні ідеї та стратегічні завдання педагогічних колективів були невід'ємною частиною, так би мовити, макро-педагогічної роботи. Поступово такі форми організації та мобілізації педагогічних зусиль ставали все менш популярними, адже робота над ними зводилась переважно до помпезного оголошення, тихого, непомітного і формального виконання та скромного звітування. Іншими словами – це був черговий підхоплений високий і авторитетний почин, хоч обов'язковий, але не зрозумілий і не забезпечений матеріально та організаційно на місцях. Але як і у всякій байці тут, безперечно,

є мораль: **ідея колективних стратегічних зусиль педагогічного колективу, хоч ще практично ніде й ні ким не доведена до логічного завершення, але корисна і при правильній організації може бути дуже продуктивною.**

Як уже згадувалоуь в першому розділі, крім проблем, які присутні при вивченні інформатики, не можна не говорити ще про одну, загальну проблему освіти – проблему зниження мотивації навчання та якості освіти. Питання мотивації навчальної діяльності для досягнення високого рівня якості навчального процесу і освіти в цілому є ключовим, тому його дослідження наведемо в окремому додатку (додаток 3). Відповідаючи на питання: яким може бути колективне завдання педагогічного колективу сьогодні, слід висувати **завдання піднесення якості навчання та рівня знань.** Особливо актуальним це стало у зв'язку із введенням зовнішнього незалежного оцінювання, яке має вирішальну роль при вступі випускників до вищих навчальних закладів та лінії на здобуття практично поголівної вищої освіти.

Навчальний процес нерозривний з виховною роботою, думка про можливість «чистого» виховання помилкова. Більше того, воно, як самоціль, не має сенсу, бо сучасне суспільство чим далі стає все більш прагматичним. Часи, коли ідеологічна складова у формуванні особистості була вирішальною, минули. На їх зміну прийшла нова, постромантична епоха. У сучасних школярах поєднуються байдужість та інфантильність із амбіційністю та практичністю. Тому не варто сподіватись, що гуманізація, демократизація чи лібералізація навчально-виховного процесу створить нове, сучасне покоління. У наш час свідомість людей об'єктивно дрейфує від моральних законів та філософських категорій зразка «совість» до відвертого визнання жорстоких законів природи зразка «боротьби за виживання». Це означає, що знання та практичні навички, які покликані приносити матеріальну користь у майбутньому, сьогодні більш цінні, ніж порядність і вихованість. Констатувати це нелегко, але факти – річ незаперечна.

Така сувора, але справедлива реальність приводять до висновків:

- 1) Навчання без виховання не ефективне.
- 2) Виховання – це сукупність прийомів, які полегшують навчання та позитивне ставлення до праці.
- 3) Виховання можливе тільки у поєднанні з навчанням та працею.

Погодившись із такими висновками, слід задуматись над суттю та масштабами виховного компонента у навчально-виховному процесі. Адже і в минулі часи не приносило успіху на уроках розв'язування задач виховного характеру. Сьогоднішнім учням слід пропонувати завдання, які розвивають компетентність, впевненість, практицизм і водночас привчають до оптимізму та реалізму.

Якщо бути відвертим і не обманювати себе, треба визнати, що дванадцятибальна система оцінювання мало змінила мотивацію до навчання, вона подібна інфляції, коли коштів більше, але варті вони менше. Учні не стали відноситись до зароблених балів, як до досягнень, які можна примножити,

вони розуміють, коли ставлять 3 бали, треба випросити ще один бал, щоб вискочити з небезпечної зони, адже 1-3 бали – це та ж колишня двійка.

Якщо порівнювати стан навчально-виховного процесу у школі з економікою країни, то доречні паралелі: ВВП – якість навчання; валюта – оцінювання.

Відомо, що *девальвація* (знецінювання валюти) напряду пов'язана із станом економіки. На початкових етапах девальвація може бути реальною але не визнаною урядом і національним банком. Для оздоровлення економіки часто проводиться регулювання курсу валюти цими установами, тобто девальвація та ревальвація.

Стан навчально-виховного процесу в школі також завжди має *реальний* і *офіційний* рівні. Головним показником навчально-виховного процесу є **якість навчання**²⁷, тобто реальне *процентне вираження відношення засвоєного навчального матеріалу до вивченого*.

Збереження та підвищення якості навчання – це головне довгострокове стратегічне і водночас постійне тактичне завдання кожної школи.

У кінцевому результаті якість навчання трансформується в освіту, яка за висловами багатьох відомих людей, зокрема Макса Лауе «є те, що залишається, коли вивчене забуте».

Одним із найважливіших тактичних інструментів тут може бути **політика оцінювання**. Вона повинна ґрунтуватись виключно на інструкціях МОН України та встановлених навчальними програмами критеріях оцінювання. Проте на практиці досягти абсолютної точності оцінювання неможливо, адже самі критерії оцінювання здебільшого дають лише приблизні інструкції, а сам предмет оцінювання – знання часто буває не чітко вираженим. До складу оцінки самої часто корисно включати виховний та заохочувальний компоненти: за однакову відповідь сильнішого учня інколи можна оцінювати нижче, ніж слабшого. Часто виникають форс-мажорні ситуації (карантин, передсвятковий період, пропуски уроків, коли проводяться заліки і виставлення тематичних оцінок, навчальні та оздоровчі поїздки груп учнів тощо), які вступають у суперечність із термінами виставлення оцінок і не завжди дозволяють дотриматись об'єктивності в оцінюванні. Накопичення таких явищ, поєднане (чого гріха таїти) з бажанням приховати реальний стан справ, приводить до *девальвації оцінювання*. Значну шкоду для об'єктивності оцінювання приносять і часто позбавлені сенсу розхожі псевдо-гуманістичні уявлення, підкріплені іменами авторитетних педагогів, наприклад: «учитель, що ставить негативну оцінку, розписується у власній неспроможності», «немає не здібних учнів, є байдужі або некомпетентні учителі» тощо.

Під *девальвацією оцінювання* слід розуміти *зниження об'єктивності оцінювання*, зокрема завищення та формальне виставлення оцінок.

²⁷ Не плутати із терміном «якість навчання», що використовувалась у школах, як процент учнів, що навчаються на «5» та «4» в старій системі оцінювання, або процент учнів, які мають достатній та високий рівні навчальних досягнень згідно 12-бальної шкали оцінювання. На відміну від колишнього змісту поняття «якість навчання», ми жодним чином не будемо прив'язувати якість навчання в новому значенні із виставленими оцінками, адже оцінки чи бали завжди носять суб'єктивний характер, тому не відображають справжній стан якості навчання.

Не треба боятись «показати» реальний стан якості навчання, тим більше, що він сьогодні вкрай низький практично в усіх середніх навчальних закладах України через нескінченні й непродумані реформи, брак коштів на освіту, корупцію в цій галузі та зміну вектора соціального замовлення на якісну освіту. У багатьох школах, які розміщені навколо великих міст, особливо мегаполісів існують додаткові об'єктивні причини для його зниження, наприклад, відтік учнів старшої школи до більш престижних навчальних закладів, які поряд із загальною освітою дають кращі стартові умови для вступу у вищі навчальні заклади.

У зв'язку з політичною нестабільністю в країні, яка особливо гостро спостерігається протягом останнього десятиріччя, на всіх рівнях стало модно говорити про особливий вид корупції - корупції в освіті, яка начебто досягла надзвичайно великих розмірів. Але вона, як і у попередні часи, все ж присутня у нашому житті. Якщо говорити про корупцію саме в освіті, то вона полягає зовсім не в тому, що педагоги отримують хабарі. Одним з її типових і розповсюджених проявів є саме девальвація оцінювання. Особливо прикро тут те, що школа, як суб'єкт такої корупції, не одержує нічого в матеріальному плані для свого розвитку, вона діє, як ланка корупційної схеми, прибутки з якої стікаються на верхні щаблі освітньої і навіть суспільної драбини. А школа таким чином «наживає» собі лише зайві проблеми, які ще більше посилюють проблему якості навчально-виховного процесу в майбутньому. Адже, якщо продовжити аналогії з економікою, за прорахунки обов'язково доведеться платити. Таке «золоте правило» життя.

Як же перевести учнів на «госпрозрахунок», щоб вони самостійно давали собі реальну оцінку, прогнозували успіхи та планували їх збільшення? Можливостей дуже багато, корисними в цьому можуть бути ІКТ та інформатика. Розглянемо це питання детальніше.

РОЗДІЛ V

V.1 ПОШУКИ ШЛЯХІВ ПОДОЛАННЯ ПРОБЛЕМ В ДОСЯГНЕННІ ВИСОКОГО РІВНЯ ЯКОСТІ НАВЧАННЯ

Виявлення проблем лише тоді буває корисним, коли воно поєднане із пошуком реальних шляхів їх подолання.

Безперечно, що пропозиція негайних «кавалерійських наскоків» для підвищення якості навчання є демагогічною. Процес подолання проблем навчання повинен мати чітку стратегію і правильно та вчасно вибрану тактику. Питання стратегії – це об'єкт окремого великого дослідження, в рамках короткого аналізу можна запропонувати лише окремі стратегічні напрямки. Одним із них повинно стати встановлення реального співвідношення між власне навчальним процесом, тобто виконанням програм і досягненням освітніх стандартів з одного боку та позакласної навчальної роботи, зокрема: конкурсів, олімпіад, роботи секцій МАН тощо.

Для того, щоб зрозуміти суть певного явища, часто буває необхідно вийти за його межі і подивитись із зовні. Такі приклади відомі з історії пі-

знання природи засобами математики, фізики, філософії і т.д. Розуміння механізмів динаміки якості навчання та освіти також не можна досягти, маючи під руками лише виставлені оцінки в класному журналі чи звітній таблиці про навчання. Нижче описано один із можливих методів розв'язання проблеми якості навчання, що ґрунтується на застосуванні (ІКТ), зокрема аналізу з допомогою електронних таблиць в середовищі Excel, який далі називатимемо **Методом Аналізу ЯКості навчання (МАЯК)**.

Проте головне в цьому методі не застосування ІКТ, а параметри оцінювання, які на відміну тих, що є у відомостях про успішність, не кількісні, а якісні. У повсякденній навчальній роботі учителі, безперечно, завжди мають їх на увазі, але зовсім не підключають для досягнення ефекту навчання. Коли виставляються оцінки по навчальних досягненнях – це лише фіксація локальних результатів, а потрібно мати глобальну, узагальнену динаміку процесу зміни якості навчання. **Коефіцієнт якості навчання**, про який ітиметься нижче, *буде визначатись зовсім без використання виставлених оцінок, навпаки, з його допомогою з'явиться можливість впливати на майбутні навчальні досягнення, виражені в балах.*

Детальніше питання визначення та методика застосування у педагогічній практиці поняття якості навчання розглянемо на прикладі нижче наведеної таблиці, яку ще в листопаді 2010 року складали учні 11-б класу Щасливського НВК під час вивчення теми «Аналіз засобами Excel».

За основу розрахунків тут візьмемо поточний стан навчання випускного класу з предмету «інформатика» згідно класного журналу трьохрічної давності (щоб не було претензій з боку фігурантів). Виберемо також період, коли у класі було проведено 10 уроків інформатики (для простоти обчислень та перевірки достовірності результатів). Як видно із таблиці 1 (див. нижче), основними параметрами для оцінювання якості навчання з інформатики взято:

1. Коефіцієнт якості відвідування (відношення відвіданих уроків до загальної кількості проведених уроків, позначимо ***Qv*** від ***Quality of visiting*** – *якість відвідування*);

2. Коефіцієнт здібностей до даного предмету²⁸ (позначимо ***Ab*** від ***Abilities*** – *здібності*);

3. Коефіцієнт активності на уроках (***Al*** від ***Activity at lessons*** – *активність на уроках*);

4. Коефіцієнт якості самостійної роботи над домашніми завданнями (***Th*** від ***Training of a house*** – *навчання дома*);

²⁸ Заперечувати, що однаково високих здібностей у всіх учнів з усіх предметів не буває можуть лише некомпетентні, або необ'єктивні особи.

5. Коефіцієнт додаткового навчання з вибраного предмета (*At* від *Additional training* – додаткове навчання).

таблиця V.1

A	B	C	D	E	F	G	H	I
Якість навчання з інформатики 11-Б класу за 10 уроків								
№	Прізвище	к-ть уроків	пропущено	здібності	актив. на уроці	робота дома	додатково	якість навчання
1	Бубенко А.	10	1	0,65	0,6	0,5	0,1	6,6
2	Бурківський Є.	10	5	0,5	0,2	0,2	0	3,4
3	Вишиван О.	10	3	0,58	0,7	0,7	0,1	6,7
4	Власюк В.	10	4	0,35	0,2	0,1	0	3,0
5	Гнип А.	10	0	0,75	0,8	0,6	0,4	8,5
6	Горецька Є.	10	3	0,6	0,4	0,3	0,1	5,0
7	Гусаков Ю.	10	7	0,5	0,1	0,1	0	2,4
8	Єфімов І.	10	4	0,66	0,5	0,5	0,1	5,7
9	Єлісєєв В.	10	2	0,66	0,5	0,5	0,1	6,1
10	Зінков О.	10	0	0,67	0,8	0,7	0,1	7,8
11	Карпінський Ю	10	1	0,64	0,3	0,2	0	4,9
12	Катана Н.	10	2	0,3	0,1	0,1	0	3,1
13	Кирильчук А.	10	2	0,3	0,3	0,1	0	3,6
14	Кириєнко Д.	10	2	0,6	0,6	0,6	0,1	6,5
15	Коваль Д.	10	0	0,9	0,6	0,45	0,2	7,6
16	Курінна Т.	10	2	0,01	0,3	0,3	0,1	3,6
17	Сидорчук К.	10	0	0,5	0,5	0,25	0,2	5,9
18	Шкляр С.	10	0	0,95	0,8	0,7	0,2	8,8
Середнє значення якості навчання класу								5,5

Якість навчання – це динамічний показник, який залежить від більшої, ніж тут вибрано, кількості параметрів і у кожен окремий момент може бути результатом складного поєднання параметрів та збігу обставин. Але, як і в будь-яких дослідженнях, закономірності зміни якості навчання можна спостерігати, користуючись обмеженим числом найважливіших параметрів, вибраних вище.

Найголовніше і майже беззаперечне те, що якість навчання, або коефіцієнт якості навчання (позначимо *Qt* від *Quality of training* – якість навчання) залежить приблизно прямо пропорційно від кожного із параметрів 1-5 і має властивість адитивності по відношенню до них, тому може обчислюватись за формулою:

Для достатньої об'єктивності визначення *Qt* слід мати по можливості точніші значення параметрів *Qv*, *Ab*, *Al*, *Th* та *At*. Для цього необхідно яко-

$$Q_t = \frac{(Q_v + A_b + A_l + T_h + A_t) \cdot 12}{5}. \quad (1)$$

мога точніше встановити зміст кожного коефіцієнта. Якщо значення *Qv* абсолютно точне, бо виставляється згідно класного журналу, то решта параметрів носять у тій чи іншій мірі суб'єктивний характер.

Дійсно, як точно визначити здібності учня до вибраного предмета²⁹ (*Ab*)? Адже вони бувають не тільки виявленими а й прихованими, потенціальними чи розвинутими, у зв'язку з чим учня можна недооцінити або навпаки – переоцінити. Для розв'язування питання про якість навчання доцільно корис-

²⁹ Доцільно визначати коефіцієнт здібностей до кожного предмету окремо, хоча можна користуватись і узагальненим коефіцієнтом здібностей.

туватись тільки виявленими і на даний момент розвинутими здібностями, адже приховані, тобто ще не виявлені чи не розвинені здібності не можуть суттєво впливати на **Ab**. Домовимось під здібностями розуміти комплекс інтелектуальних складових: **ерудиція** (набір здобутих знань та умінь), **інтуїція** (здатність генерувати ідеї), **креативність** (вміння творчо поєднувати ерудицію та інтуїцію), **концентрація уваги** (здатність виділяти головне та зосереджуватись на ньому) і **пам'ять** (зорова, слухова, швидка і т.д.).

Якщо дотримуватись не емпіричного, а наукового підходу, то для визначення коефіцієнту здібностей з даного предмета **Ab** треба застосувати так зване вхідне тестування, загальне чи з окремо вибраного предмету, яке додатково дозволить також встановити достовірну інформацію про якість базових знань на момент тестування. Ми пропонуємо тестування на виявлення здібностей проводити з допомогою відомих психологічних тестів на короткочасну пам'ять, концентрацію уваги тощо, які легко реалізувати засобами табличного процесора (див. додаток). Але можна виявляти коефіцієнт **Ab** колегіально, врахувавши думку щодо нього кожного, або хоча б двох із трьох незалежних експертів: самого учня, учителя та батьків, причому мати на увазі тільки виявлені на даний час. На здібностях, як на фундаменті, ґрунтуються реальні знання, уміння та навички, хоча їх якість досягається не лише за рахунок здібностей.

У своїй практиці для виявлення коефіцієнта здібності ми користуємось комплексним методом, який складається із етапів *прикидки, збору даних, тестування, аналізу, узагальнення та колегіального узгодження*.

Параметри **Al** (активність на уроках) та **Th** (якість виконання домашніх завдань) цілком об'єктивно може встановити учитель, але бажано обговорити й узгодити його з учнем, щодо параметра **At** учень та учитель також можуть легко дійти згоди в результаті бесіди. Потреби формалізувати параметри з допомогою підрахунку кількості відповідей на уроці, кількості правильних із них, похвилинного хронометражу класної чи домашньої роботи, на наш погляд, немає, адже саме поняття якості навчання у вибраному тут сенсі досить абстрактне і не може бути абсолютно точним.

Для повноти аналізу методики визначення якості навчання проаналізуємо формулу (1). Як видно із таблиці, кожен її компонент представлений у вигляді долі від одиниці, або у процентному вимірі, якщо помножити його на 100, тому для ідеального учня всі параметри будуть рівними одиниці. Середнє арифметичне, помножене на 12 переводить коефіцієнт якості навчання (**Qt**) в 12-и бальну систему. Формулу (1) можна легко змінити, взявши іншу кількість параметрів (2). Наприклад, можна додатково ввести коефіцієнт базового рівня знань **Bl** (**Base level** – базовий рівень).

Як уже сказано вище, коефіцієнт **Bl** може бути об'єднаний із коефіцієнтом **Ab** під час вхідного тестування. Тому описаний метод можна вважати

$$Q_t = \frac{(P_1 + P_2 + \dots + P_n) \cdot 12}{n} \quad (2)$$

універсальним.

Безперечно, можна вибрати якусь іншу формулу для обчислення Q_t , головне, щоб вона також наочно демонструвала стан якості навчання й давала можливості для її дослідження. Однією з них може бути формула (3):

Вона побудована на мультиплікаційному принципі врахування параметрів. Продемонструємо і обговоримо цю формулу нижче (див. табл. 2):

$$Q_t = \frac{(P_1 \cdot P_2 \cdot \dots \cdot P_n) \cdot 1000}{n} \quad (3)$$

Таблиця 2

№	Прізвище	Q_v		Ab	Al	Th	At	Q_t (ад.)	Q_t (мул.)
1	Бубенко А.	10	1	0,65	0,6	0,5	0,1	6,6	17,6
2	Бурківський Є.	10	5	0,5	0,2	0,2	0	3,4	0,0
3	Вишиван О.	10	3	0,58	0,6	0,5	0,1	6,0	12,2
4	Власюк В.	10	4	0,35	0,2	0,1	0	3,0	0,0
5	Гнип А.	10	0	0,75	0,8	0,6	0,4	8,5	144,0
6	Горецька Є.	10	3	0,6	0,5	0,4	0,1	5,5	8,4
7	Гусаков Ю.	10	7	0,5	0,1	0,1	0	2,4	0,0
8	Єфімов І.	10	4	0,66	0,9	0,7	0,1	7,1	24,9
9	Єлисєєв В.	10	2	0,66	0,75	0,7	0,1	7,2	27,7
10	Зінков О.	10	0	0,67	0,85	0,75	0,1	8,1	42,7
11	Карпінський Ю.	10	1	0,64	0,2	0,2	0	4,7	0,0
12	Катана Н.	10	2	0,3	0,1	0,1	0	3,1	0,0
13	Кирильчук А.	10	2	0,45	0,3	0,4	0,1	4,9	4,3
14	Кирієнко Д.	10	2	0,6	0,5	0,4	0,1	5,8	9,6
15	Коваль Д.	10	0	0,9	0,5	0,45	0,2	7,3	40,5
16	Курінна Т.	10	2	0,01	0,01	0,2	0,1	2,7	0,0
17	Сидорчук К.	10	0	0,9	0,6	0,45	0,2	7,6	48,6
18	Шкляр С.	10	0	0,95	0,7	0,7	0,2	8,5	93,1
Середнє значення якості навчання класу								5,7	

Як видно з останнього стовпчика таблиці 2, де обчислено значення Q_t за формулою (3), для збільшення значень якості навчання у балах множник 12 довелось замінити множником 1000. Незважаючи на це, формула (3) не придатна для використання, адже значення, обчислені за нею, явно не правдоподібні. Наприклад, для учнів Губенко А. та Гнип А. відношення між балами дуже відрізняються, зокрема $6,6/8,5 = 0,77$, а $17,6/144 = 0,12$. Для учнів Карпінського Ю. та Катаної Н. Q_t , обчислене за формулою (3) взагалі дорівнює нулю, що неправильно. Причина в тому, що для цих учнів коефіцієнт At дорівнює нулеві. Це і є основною причиною непридатності формули (3). Надалі вибрану для обчислення коефіцієнта якості навчання формулу будемо називати **робочою формулою**.

Хто, коли і як може застосовувати описаний метод? Легше відповісти, кому, коли і при яких обставинах він може бути зайвим чи зашкодити. Ось орієнтовний, далеко не повний перелік застосувань:

- **Учитель-предметник** для аналізу якості навчання учня, групи, класу, тематично, посеместрово, щорічно, на перспективу;

- **Окремий учень** в будь який момент навчального процесу або при очікуванні під-сумкового балу;
- **Класний керівник** при щомісячній атестації класу, при підготовці до батьківських зборів, при прогнозуванні та плануванні якості навчання;
- **Адміністрація школи** у всіх згаданих вище ситуаціях;
- **Батьки** при визначенні навчальних успіхів свого учня.

Завдяки поширенню комп'ютерів та можливостям табличного процесора *Excel* описаний метод доступний практично завжди й усім учасникам навчально-виховного процесу. Метод може бути застосований для окремого учня, класу і навіть цілої школи, для окремо вибраного предмета, для вибраної групи предметів чи узагальнено для всіх предметів. Його можна застосовувати одноразово, епізодично, вибірково чи постійно, для встановлення статистичних показників, тобто для спостереження за рівнем навчання та його якістю, а також для прогнозування і планування. **Можливості прогнозуван-**

J2		fx = ((C2-D2)/C2+F2+G2+H2+I2)/5*12								
	A	B	C	D	E	F	G	H	I	J
			к-ть уроків	пропущено	введення	здібності	актив. на уроці	робота дома	доплатково	якість навчання
1	№	Прізвище								
2	11	Карпінський Ю.	10	1	0,9	0,44	0,2	0,2	0,02	4,2
3										

мал.1

ня та планування якісних показників навчання з допомогою описаного методу можна цілком правомірно вважати новаціями³⁰ в опрацюванні результативності навчання.

Вище описана лише одна із численних можливостей впровадження аналітичних методів у навчально-виховний процес. Для пояснення розглянемо приклад з окремим учнем, взявши учня 11-б класу Карпінського Ю. (фрагмент електронної таблиці див. на мал. 1).

З малюнка 1 видно, що пропуски уроків не впливають суттєво на якість навчання, але коефіцієнт якості навчання Q_t , що дорівнює 4,9 балів, не може задовольняти випускника. Де можна знайти резерви? Перш за все це активність на уроці та робота дома. Нехай учень хоче досягти в атестаті оцінку 7 балів з інформатики.

Як видно з малюнка 2, досить підвищити інтенсивність роботи на

	A	B	C	D	E	F	G	H	I	J
			к-ть уроків	пропущено	введення	здібності	актив. на уроці	робота дома	доплатково	якість навчання
1	№	Прізвище								
2	11	Карпінський Ю.	10	1	0,9	0,44	0,8	0,8	0,02	7,1

мал.2

³⁰ Не плутати з інноваційними методами!

уроці та дома до 0,8 (80%), як коефіцієнт якості навчання Q_t стане рівним 7,1 бала.

Діаграма, зображена на малюнку 3, наочно ілюструє резерви для підвищення коефіцієнта Q_t . При нормальному стані речей коефіцієнт відвідування Q_v , як і всі інші, не може перевищувати 0,2. При коефіцієнті здібностей Ab , який дорівнює 0,25, тобто вище середнього, за рахунок активності на уроках та дома, а також хоч невеликих зусиль, спрямованих на додаткову роботу з предмета (консультації та перездачі окремих тем, курси при вищому навчальному закладі, уроки репетитора тощо), Карпінському Ю. легко досягти якості навчання значно вище за 7,1, аж до 10 балів.

При вмілому використанні подібного аналізу класному керівникові легко давати конкретні поради учневі та його батькам щодо покращення результативності навчання, а батьки з його допомогою легко зможуть скоригувати свої фінансові зусилля та навчальну роботу свого сина для успішного вступу до вищого навчального закладу.

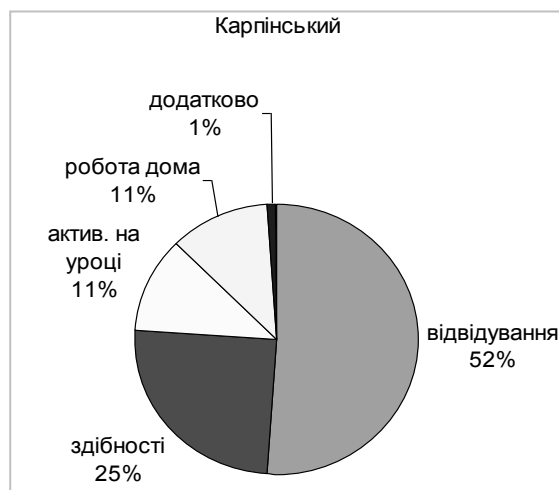
Якщо створити зведені таблиці якості навчання для кожного класу, то з їх допомогою легко виявляти кількість потенціальних відмінників, кандидатів на відзначення свідоцтвами з відзнакою та Золоті і Срібні медалі, прогнозувати результати майбутніх ЗНО, робити вчасну корекцію навчально-виховного процесу. Адміністрація школи зможе без формалізму за цілий рік, а то й два наперед планувати кількість претендентів на відзнаки тощо.

Іншими словами, застосувавши ще раз аналогію, але тепер уже з військовою справою, описаний метод використання коефіцієнта якості навчання в дії, на зразок військових штабних ігор, може допомогти вибрати правильну тактику та стратегію в реальній справі контролю за якістю навчально-виховного процесу.

Можна погодитись, що тут описано ще далеко не всі можливості методу визначення якості навчання з допомогою ІКТ.

Єдиний параметр – Ab (здібності до предмета) виглядає, як статичний, адже основним компонентом здібностей вважається їх генетична (вроджена) складова. Проте вони також можуть розвиватись в результаті систематичної роботи. Таким чином, використовуючи метод визначення якості навчання, можна розвивати навіть здібності учня.

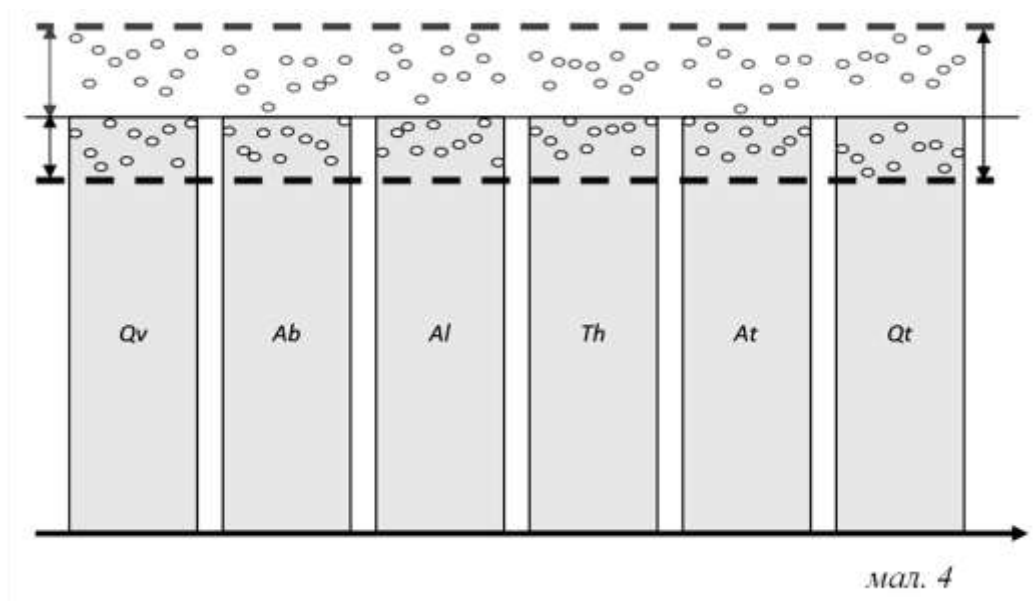
Можливим недоліком покладеного в основу методу принципу адитивності в формулі обчислення Q_t (див. формули (1) та (2)), є рівність вибору долі кожного компонента. Але цей можливий недолік в той же час є перевагою такого вибору формули. Це впливає із того, що при значенні кожного компонента 1 (100%) одержаний коефіцієнт Q_t також дорівнюватиме 1.



мал.3

$$Q_t = \frac{(j_1 \cdot P_1 + j_2 \cdot P_2 + \dots + j_n \cdot P_n) \cdot 12}{n} \quad (4)$$

Уникнути вказаного можливого недоліку можна, встановивши у формулу (3) систему коефіцієнтів j_i , де $1 \leq i \leq n$ (див. формулу (4)):

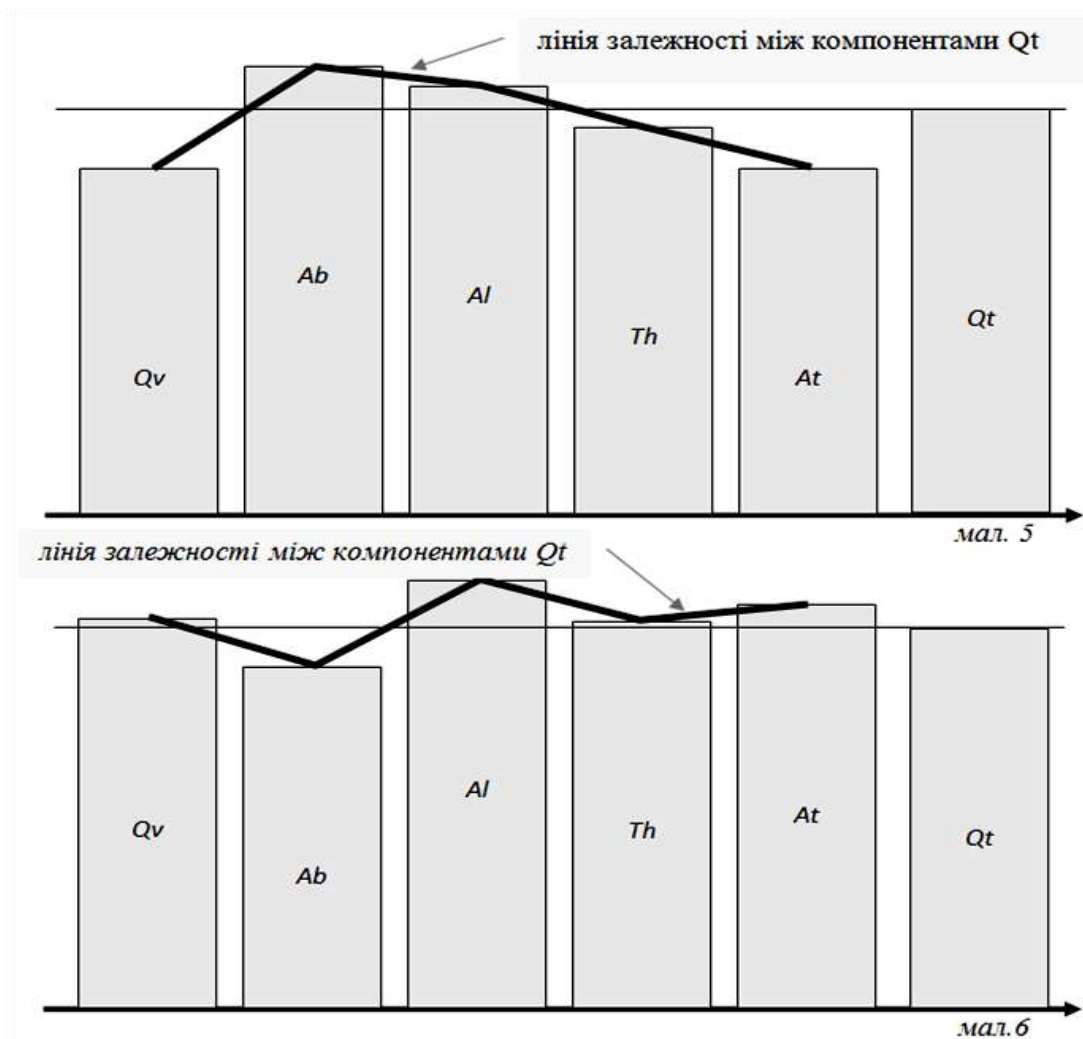


Проте, краще такі коефіцієнти не долучати, адже, по-перше, їх природу необхідно додатково досліджувати експериментальним шляхом, причому, вони можуть виявитись змінними, по-друге, їх вплив на обчислення **Qt** не може бути значним. (для $n = 5$ див. мал. 4).

На малюнку всі компоненти взято, як і у формулі, рівними (по 20% кожен). Якщо при статистичних дослідженнях формули взяти реальні відхилення в сторону збільшення (показано синім кольором, а максимальне відхилення позначено h_t), та в сторону зменшення (показано червоним кольором, а максимальне відхилення позначено h_b), то вони будуть корелювати навколо середнього значення і максимальне відхилення може виявитись $h_t + h_b = h_d$. Слід очікувати, що h_d не буде значним по відношенню до величини кожного компонента.

Розробку остаточної формули підрахунку якості навчання можна до- ручити експертній комісії, до складу якої можуть входити представники ад- міністрації школи та голови предметних методичних об'єднань. Як видно із таблиці 3, наведеної нижче, можливі коефіцієнти при параметрах **Qv**, **Ab**, **Al**, **Th**, та **At** можуть бути на порядки меншими за 1, адже розходження **Qt** із се- местровою оцінкою коливається між 0,1 та 0,9 бала, а в середньому дорівнює 0,46 бала, або всього 3,84%. При цьому збільшення будь-якого коефіцієнта можливе лише за рахунок зменшення одного або кількох серед інших чоти- рьох. Проілюструвати сказане можна з допомогою двох наступних малюнків (мал.5-6). На них показано порівняльні діаграми складу якості навчання для двох різних учнів. У одного, в силу індивідуальних особливостей (високий показник здібностей та пов'язана з ним здатність опановкувати навчальним матеріалом у класі – мал. 5).

Проілюструвати сказане можна з допомогою двох наступних малюнків (мал.5-6). На них показано порівняльні діаграми складу якості навчання для двох різних учнів. У одного, в силу індивідуальних особливостей



(високий показник здібностей та пов'язана з ним здатність опановкувати навчальним матеріалом у класі – мал. 5).

Інший не відзначається високими здібностями, але «бере» за рахунок працелюбства у класі, вдома та працюючи з предмету додатково (мал.6).

Таким чином, виходячи із динамічної природи складу коефіцієнта якості навчання Q_t та неможливості виключити випадкові компоненти впливу на нього, можна визнати цілком прийнятною робочу формулу (1).

V.ІІ. ОБҐРУНТУВАННЯ МЕТОДУ АНАЛІЗУ ЯКОСТІ НАВЧАННЯ

Як уже було сказано вище (в п. II), найбільш сумнівним і водночас найважливішим питанням в методі аналізу якості навчання (МАЯК) є вибір робочої формули (нагадаємо, що ми зупинились на (1)). Тому спробуємо дослідити робочу формулу на даних, одержаних із журналу обліку успішності, тобто, перевіримо цю формулу на придатність. Для цього порівняємо таблицю 1 із наведеною нижче таблицею 3, яка відображає якість навчання 11-Б класу на кінець I семестру. Припустивши, що оцінки, виставлені за I семестр,

об'єктивні абсолютно, легко побачити різницю між ними та обчисленим коефіцієнтом якості (табл.3).

Якщо врахувати, що оцінка може бути тільки цілим числом, то семестрові оцінки вищі на 1 бал у одного учня (Кириєнко Д.) і нижчі на 1 бал у 6 учнів (Бурківський Є., Горецька Є., Єфімов І., Єлисеєв В., Кирильчук А., Сидорчук К.). Отже семестрові оцінки не відповідають фактичній якості навчання на 1 бал у 7-и учнів з 18, тобто у 38,8%.

ЯКІСТЬ НАВЧАННЯ З ІНФОРМАТИКИ 11-Б КЛАСУ ЗА І СЕМ. таблиця V.3

№	Прізвище	к-ть уроків	пропущено	здібності	активність	робота дома	додатково	якість навч.	I семестр	різниця	якість навчання
1	Бубенко А.	17	3	0,65	0,6	0,5	0,1	6,4	6	0,4	6,6
2	Бурківський Є.	17	7	0,5	0,2	0,2	0	3,6	3	0,6	3,4
3	Вишиван О.	17	5	0,58	0,7	0,7	0,1	6,7	7	0,7	6,7
4	Власюк В.	17	5	0,35	0,2	0,1	0	3,3	3	0,3	3,0
5	Гнип А.	17	0	0,7	0,8	0,6	0,4	8,4	8	0,5	8,5
6	Горецька Є.	17	6	0,6	0,4	0,3	0,1	4,9	4	0,9	5,0
7	Гусаков Ю.	17	14	0,5	0,1	0,1	0	2,1	н/а		2,4
8	Єфімов І.	17	8	0,6	0,5	0,4	0	4,9	4	0,9	5,7
9	Єлисеєв В.	17	3	0,6	0,5	0,4	0	5,6	5	0,6	6,1
10	Зінков О.	17	1	0,6	0,7	0,7	0,1	7,3	7	0,3	7,8
11	Карпінський Ю.	17	2	0,64	0,3	0,2	0	4,9	5	0,1	4,9
12	Катана Н.	17	5	0,3	0,1	0,1	0	2,9	3		3,1
13	Кирильчук А.	17	9	0,3	0,3	0,1	0	2,8	2	0,8	3,6
14	Кириєнко Д.	17	4	0,6	0,6	0,6	0,1	6,4	7	0,6	6,5
15	Коваль Д.	17	0	0,9	0,6	0,45	0,2	7,6	8	0,4	7,6
16	Курінна Т.	17	7	0,01	0,3	0,3	0,1	3,1	3	0,1	3,6
17	Сидорчук К.	17	2	0,5	0,5	0,25	0,2	5,6	5	0,6	5,9
18	Шкляр С.	17	1	0,95	0,8	0,7	0,2	8,6	9	0,4	8,8
Середнє значення якості навчання класу								5,1		5,5	

При врахуванні розряду десятих максимальна невідповідність між **Q_t** та семестровою оцінкою (0,9 бала) виявляється лише у двох учнів. При цьому слід врахувати, що у деяких випадках різниця в 1 бал допустима, наприклад, при тематичних «4» та «5» за семестр можна ставити «5», а при тематичних «5» та «4» правильніше виставляти за семестр «4», адже вона у значній мірі є підсумковою, причому часто друга тема важливіша, а знання, уміння та навички першої теми під час вивчення другої повинні бути підтверджені.

Очевидно, що при заміні формули (1) формулою (4) може бути досягнута ще вища об'єктивність. Але навіть за формулою (1) до виявлених похибок ми прийшли з припущення, що семестрові оцінки вірні, насправді це буває не завжди так, як з об'єктивних, так і з необ'єктивних причин. У випадку, коли обидва показники не абсолютно точні, істину слід шукати посередині, тобто *описаний метод дає цілком прийнятне відхилення від істини: повна відповідність – понад 60%; виявлене відхилення не вище 1 бала – 38% учнів, реальне відхилення не вище 1 бала, – значно менше третин учнів.*

Отже, підводячи підсумки щодо обґрунтування точності вибраної формули, можна констатувати, що вона залежить від якості визначення її параметрів. Проте часто навіть серйозні наукові висновки робляться, посилаючись на звичайне тестування або анкетування. Хто сумнівається, що так званий коефіцієнт інтелекту, тобто ***IQ***, який одержують виключно в результаті швидкого тестування, сумнівний? Але навіть в інструкції для роботи по визначенню ***IQ*** є вимога відповідати на питання швидко, не задумуючись, не сумніваючись даючи правдоподібні відповіді, що першими спали на думку.

V.ІІІ. МЕТОДИЧНІ ПОРАДИ ЩОДО ВИКОРИСТАННЯ МЕТОДУ ЗАСТОСУВАННЯ КОЕФІЦІЄНТА ЯКОСТІ НАВЧАННЯ У НАВЧАЛЬНО-ВИХОВНОМУ ПРОЦЕСІ

Ефект використання методу МАЯК буде відчутним, якщо його використовувати системно, постійно і всеохоплююче, тому розробку потрібної технології (надалі називатимемо її ***технологією МАЯК***) природно вважати *колективним проектом*, у який слід включити цільові групи педагогічних працівників:

- ***адміністративно-координаційна група*** (керівник проекту, експерт-консультант, відповідальні за виконання);
- ***експериментально-координаційна група*** (з представників методичного об'єднання класних керівників);
- ***експертна група*** (з голів предметних методичних об'єднань учителів окремих предметів).

Очевидно, що робота над проектом буде тривалою в часі і може зайняти цілий семестр а то й навчальний рік. Для координації дій і встановлення нормативних правил необхідно спланувати виконання проекту, передбачити виробничі наради, семінари-практикуми та винесення питання розробки проекту і створення технології на засідання педагогічної ради.

Роботу над проектом орієнтовно можна розбити на етапи:

1. ***інформаційний етап*** (формулювання, роз'яснення та популяризація ідей проекту, на сьогодні можна вважати частково виконаним);
2. ***організаційний етап*** (створення цільових груп, розгляд проекту на педагогічній раді, планування та нормативне оформлення);
3. ***імплементаційний*** (виконавчий) ***етап (Implementation – виконання)***;
4. ***узагальнюючий етап*** (цей етап не можна називати підсумковим, адже в результаті виконання проекту повинна з'явитись обґрунтована

технологія МАЯК. Надалі ця технологія може стати постійною складовою реалізації навчально-виховного процесу).

Для повномасштабної реалізації методу МАЯК, переважній більшості членів педагогічного колективу необхідно мати хоча б початкові уявлення про роботу в середовищі Excel і застосовувати вибрану технологію. Для цього на організаційному етапі реалізації проекту слід провести 1-2 практикуми-тренінги по Excel.

Опишемо кожен із етапів роботи над проектом.

Інформаційний етап

Основним об'єктом, який необхідно створити для аналізу якості навчання, є електрон-на таблиця, що містить певну кількість листів із таблицями для аналізу якості навчання з кожного окремо взятого предмета та зведену таблицю якості навчання (надалі – **пакет таблиць**). Але почати потрібно із створення такої таблиці з одного предмета, на якій виконати початкове дослідження робочої формули. Цей етап роботи достатньо детально описано вище в п. II.

Для подальшого дослідження робочої формули слід провести експериментальне створення пакету таблиць для 2-3 класів. Цю роботу виконують експерт-консультант та класні керівники вибраних класів. Початкова робота класних керівників зводиться до заповнення таблиць для свого класу наведеного нижче зразка (див. мал. 4).

Інформація про здібності, активність на уроках та роботу над домашніми завданнями учнів 8-а класу

таблиця 4

	укр.мова			укр.літ.			зар.літ.			англ.			– – –			інформ.			середні		
	З	у	д	З	у	д	З	у	д	З	у	д	–	–	–	З	у	д	З	у	д
Учні																					
Андріанова А.																					
Вакулюк В.																					
Відута В.																					

Ця таблиця містить список усіх учнів класу в алфавітному порядку, для кожного з яких відведено клітинка, де фіксуються виявлені здібності (**З**), поточний стан якості роботи на уроках (**У**) та поточний стан якості підготовки домашніх завдань (**Д**) з десяти основних навчальних предметів.

Класні керівники контролюють заповнення цієї таблиці відповідними учителями, які знизу візують підписами свої записи, та подають готові таблиці експерту-консультанту, який переносить записи в пакет таблиць відповідного класу.

Після цього остаточно формується пакет таблиць на якому додатково перевіряється точність робочої формули, робляться висновки адміністративно-координаційної групи і після оперативної наради проект переходить до стадії організаційний етапу. Але ми маємо справу не тільки з колективним проектом, а й з колективною відповідальністю, тому необхідно застерегти учасників проекту про дотримання уже на інформаційному етапі високої точності та коректності даних. Не для критики, а лише у якості ілюстрації слід навести приклад, коли при заповненні таблиці (мал. 4) один з учителів у гру-

пі з 15 учнів п'ятим учням виставив у графах (З), (У) та (Д) виставив "1", що означає 100%. Можна пояснити явну помилку цього учителя недосвідченістю, тому обійдемося без прізвища, але й несвідоме спотворення вхідних даних може закреслити весь проект.

Для підтвердження думки, сформульованої в попередньому абзаці, щодо недостатньо якісного оцінювання вчителями роботи учнів, додатково наведемо дані про 8-А клас.

таблиця 5

учні	Укр. мова та літерат.	Російська мова та зар. літер.	Англійська мова	іст.орія	біологія	хімія	фізика	алгебра	геометрія	інформатика
Андріанова А.	1	0,8	0,8	0,6	1	1	0,8	0,8	0,7	0,6
Вакулук М.	0,9	0,9	0,6	0,7	0,9	0,9	1	1	0,9	0,8
Видута В.	0,8	0,8	0,6	0,6	0,7	0,6	0,7	0,8	0,8	0,66
Гармашов В.	0,8	1	0,9	0,9	1	0,5	0,7	0,6	0,5	0,5
Голяка Є.	0,9	0,8	0,6	0,5	0,6	0,5	0,4	0,6	0,5	0,55
Зацарінний Р.	1	1	0,9	1	1	1	1	1	1	0,8
Каськів І.	1	0,9	0,9	0,9	0,9	0,7	0,9	0,8	0,6	0,66
Качур А.	0,8	0,8	0,6	0,5	0,7	0,5	0,6	0,6	0,6	0,66
Кизим Н.	1	0,9	0,9	0,8	1	1	0,9	0,8	0,8	0,7
Ковальов О.	0,8	0,8	0,9	0,7	0,8	0,6	0,6	0,7	0,7	0,66
Колосок О.	0,8	0,8	0,8	0,5	0,7	0,5	0,5	0,5	0,5	0,55
Кравченко Є.	0,6	0,7	0,5	0,4	0,7	0,5	0,5	0,5	0,5	0,55
Кришевська А.	0,9	0,8	1	0,7	0,8	0,6	0,8	0,7	0,6	0,55
Крупеня О.	0,7	0,7	0,5	0,4	0,7	0,6	0,7	0,6	0,6	0,55
Кузьмич А.	1	1	0,9	1	1	0,9	1	1	1,0	0,8
Лазоренко В.	1	0,8	0,8	0,7	0,9	0,8	0,9	1	1	0,7
Малиневська О.	1	1	0	1	1	0,8	0,9	1	1	0,75
Марченко Ю.	0,7	0,8	0,9	0,8	0,8	0,7	0,5	0,5	0,5	0,6
Мельниченко М.	1	1	0,7	1	0,9	1	1	0,7	0,7	0,75
Неумивака О.	0,7	0,7	0,9	0,6	0,7	0,6	0,5	0,5	0,5	0,55
Оліховська Ю.	0,6	0,7	0,8	0,7	0,8	0,7	0,7	0,7	0,6	0,7
Остаповець Д.	0,8	0,7	0,8	0,9	0,9	0,9	1	0,8	0,7	0,85
Петренко М.	0,6	0,6	0,8	0,4	0,6	0,5	0,6	0,7	0,6	0,66
Петров Е.	0,6	0,6	0,9	0,4	0,5	0,4	0,4	0,6	0,5	0,7
Радюшкіна А.	1	1	0,4	1	1	0,8	0,8	0,9	0,9	0,7
Резніков А.	0,8	0,8	0,9	0,9	0,9	1	0,8	0,8	0,8	0,75
Романенко А.	0,6	0,6	0,8	0,7	0,7	0,6	0,6	0,5	0,5	0,66
Тарасов М.	0,8	0,6	0,9	0,7	0,6	0,4	0,5	0,6	0,6	0,6
Шкара Т.	0,7	0,7	0,7	0,6	0,7	0,7	0,7	0,7	0,7	0,75
Шкляр А.	1	1	0,7	1	1	1	1	1	1	0,75
Ярмоленко С.	0,9	1	0,9	0,9	0,9	0,6	0,8	0,9	0,8	0,66

При переведенні оцінок з 12-и бальної системи у долі від одиниці, або проценти одержиться таблиця:

таблиця 6

Бали	1	2	3	4	5	6	7	8	9	10	11	12
Долі	0,08	0,17	0,25	0,33	0,42	0,5	0,58	0,67	0,75	0,83	0,92	1,0

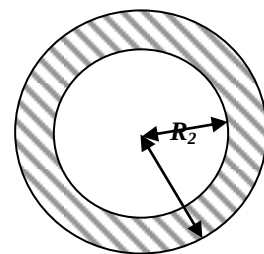
Для початку звернемо увагу на 1, тобто 100% оцінку. В учнів: Зацарінного Р., Кузьмич А., Малинєвської О., Мельниченко М., Радюшкіної А. та Шкляр А. такі оцінки переважають. Ці шестеро учнів дійсно одні з кращих у класі, але якщо вважати, що 1 (100%) – це виключно 12 балів, то чому ж тоді в класі немає шести «круглих» відмінників? Згідно звіту за I семестр у цьому класі відмінників узагалі немає, а «хорошистів» лише двоє. Думається, учителі, які виставляли 1 (100%), мали на увазі приблизно 9-10 балів, тобто у названих учнів замість одиниць повинні стояти 0,75-0,85 усього балів. Очевидно, що решта учнів оцінена також із завищенням. Чому так дивно виходить? Безперечно, учителі здебільшого не усвідомили сповна систему оцінювання і заплутались, переводячи бали 12-бальної системи у долі одиниці. Але перш за все слід позбутись хибної точки зору, що учень, який зайняв на районній олімпіаді призове місце, автоматично заслуговує найвищого балу в класному журналі. Наприклад, учениця 11-А класу Лисенко А. дійсно заслуговує з інформатики 10-11 балів (для заохочення інколи одержує навіть 12 балів). Але на районній олімпіаді через 20 хв. зрозуміла, що не зможе справитись із жодним із завдань. У той же час Єлисеєв Е., який з інформатики стабільно одержує 7-8-9 балів, а з математики ще нижче, другий рік підряд виборює I місце на районній олімпіаді, причому, із значним відривом. Правильно буде, коли в атестат він одержить 10 балів, якщо учень буде працювати до кінця навчального року. Станом на сьогодні, він багато пропускає, тому за рік може одержати нижчу оцінку. Ми ж часто, щоб дати не претендентові на медаль Грамоту за успіхи у вивченні окремого навчального предмету, ставимо йому 12 балів, інакше він не зможе, згідно відповідної інструкції одержати відзнаку.

Із сказаного вище випливає, що використовувати таблицю 4 без уважного її перегляду для аналізу якості навчання у 8-А класі неможливо. На цьому етапі зрозуміло, що значна частина учителів не вміють, або не хочуть об'єктивно оцінювати, наприклад, здібності учнів, тому спробуємо розібратись в питанні детальніше. Під час чергового заняття факультативу з інформатики 22 січня 2011 р. біля дошки колективно розбирали хід побудови математичної моделі задачі:

Визначити радіус меншого кола кільця за даними радіусом більшого кола та площею кільця.

Іншими словами, потрібно було розв'язати планіметричну задачу, доступну середнім учням 8 класу. Але йшлося не про розв'язування задачі, а лише про її колективний розбір з допомогою вчителя, що значно легше. Присутніми були учні 8-9 класів, зокрема: Остаповець Д., Ковальов О. Зазимко В. та Синяченко А. Передбачалось зробити розбір з допомогою наведених запитань та завдань:

- 1) Які позначення необхідно зробити для розв'язування задачі?
- 2) Накреслити кільце та заштрихувати його;
- 3) За якою формулою треба обчислювати площу такого кільця?



мал. 7

4) Обчислити площу кільця при $R_1 = 3\text{см}$, а $R_2 = 1\text{см}$.

5) Написати формулу для обчислення R_2 .

Але вже питання 3) поставило всіх учнів глухий кут. Після того, як усі учні зробили по кілька спроб, один з них на дошці з упевненістю написав остаточної формулу:

”Не помітивши” аж двох помилок, учитель ”підхопив” ініціативу учнів і попросив визначити площу кільця. Звичайно, вийшла неочікувана відповідь, тобто $12,56\text{см}^2$ замість $25,12\text{см}^2$. Учні, знову ж колективно, догадалися, що правильна формула повинна мати вигляд: $S = \pi(R_1^2 - R_2^2)$. Учитель, знову «не помітивши» помилки, запропонував ще раз визначити площу кільця. Звичайно в учнів вийшло $-25,12\text{см}^2$. Учні навіть не помітили, що площа виражена від’ємним числом. І лише після того, як на це звернув увагу вчитель, ... зайшли в «тупик», тобто, ні колективно, ні самотійно, але без учителя, не змогли знайти свою помилку. Так само важко було учням написати формулу, що випливає з питання 5):

$$R_1 = \sqrt{\frac{\pi R_2^2 - S}{\pi}}.$$

Інколи при розв’язуванні задачі виникає «зациклення», що природно, але ж не на такій простій задачі, яку колективно розв’язували не слабші учні 8 та 9 ліцейних класів протягом 35-40 хвилин. Був проведений аналіз роботи учнів. Вони погодились, що у 8 класі серед завдань контрольної роботи з геометрії могла бути подібна задача. Якщо припустити, що завдань такого рівня в контрольній роботі було 5, а відведено часу 45 хв., то на одну подібну задачу можна було б відвести не більше 9 хв. При опитуванні виявлено, що названі учні за I семестр з геометрії мали 8-9 балів. У той же час подібний тест говорить, що здібності з математики вони мають значно нижчі, які ледве вписуються в категорію «середній рівень», адже задача передбачає знання лише однієї загальновідомої і часто вживаної формули, потребує мінімуму елементарних алгебраїчних перетворень та містить всього два прості логічні кроки: подвійного застосування формули і вираження одного з її компонентів через інші.

Щоб завершити питання про інформаційний етап, треба додати, що було б корисно надати цей матеріал, який дає вичерпну інформацію про методику визначення та використання якості навчання, усім членам педагогічного колективу у вигляді брошур розміром А5 (альбомна орієнтація), яка поміститься на 12 сторінках, надрукованих при двосторонньому друці по дві сторінки на аркуші А4.

Враховуючи те, що здібності, як уже згадувалось, можуть бути на даний момент ще не розвинені або не повністю розкриті, щоб мати об’єктивніші дані, для визначення коефіцієнта **Ab** корисно провести 2-3 серії експрес-тестування. Особливо це потрібно, коли клас лише розпочав вивчати новий навчальний предмет та відбулось порівняно небагато уроків. Напри-

клад, інформатику в 7 ліцейному класі тільки розпочато вивчати. В ньому згідно навчального плану щотижня відводиться одна навчальна година. Протягом семестру фактично відбулось лише 15 уроків (один не проведений через відміну 6-7 уроків з приводу шкільного свята і один – через відсутність учителя, який перебував у відрядженні). Крім того, практично всі учні протягом семестру пропустили з різних причин по 1-5 уроків інформатики. Отже для виявлення реальних здібностей та якості підготовки домашніх завдань не завжди було вдосталь часу.

Організаційний етап

На цьому етапі необхідна чітка й оперативна інструктивно-роз'яснювальна робота адміністративно-координаційної групи, яка б забезпечила надалі «чистоту» та результативність проходження експерименту. Додати можна лише те, що розпочинати такий масштабний проект не варто, якщо в самої адміністративно-координаційної групи немає повного розуміння суті і значення експерименту та достатньо волі для його логічного завершення, адже саме це може мобілізувати весь педагогічний колектив.

Необхідно також скласти чіткий план, встановити оптимальні строки проходження етапів та частин експерименту, продумати ефективний контроль та своєчасні аналіз і оперативне втручання в хід виконання проекту.

Імплементаційний етап

Він передбачає перш за все створення для всіх або вибраної категорії класів пакетів таблиць, за що несуть відповідальність класні керівники. Після створення пакетів таблиць слід провести порівняльний аналіз із реальними результатами навчання за I семестр та визначити терміни повторного аналізу. Зразу після створення пакетів таблиць та проведення аналізу необхідно довести отримані результати до учнів та батьків, визначити перспективи кожного учня щодо підвищення якісного показника навчальної діяльності і узгодити з ними розміри та строки реальних досягнень.

Паралельно адміністративно-координаційна група повинна здійснювати моніторинг виконання проекту а експертна група продовжувати дослідницьку роботу над робочою формулою та методикою впливу на навчальний процес з допомогою технології Маяк³¹.

Узагальнюючий етап

На цьому етапі повинні бути готовими результати експерименту та висновки щодо користі методу і перспектив його подальшого застосування. На основі досягнутого залишиться чітко сформулювати і описати технологію МАЯК та дати методичні рекомендації щодо її застосування. Як уже згадувалось вище, цей етап не можна називати підсумковим, в майбутньому слід проводити постійну роботу над підвищенням якості навчання й освіти в цілому з допомогою технології МАЯК.

³¹ Точніше, паралельно з експериментальною розробкою технології МАЯК.

Щоб не забігати далеко наперед, можна лише запланувати одержання висновків по підсумках навчального року, ДПА для учнів 9 та 11 класів, зовнішнього незалежного оцінювання і на їх основі зробити прогнози на наступний навчальний рік та врахувати їх у річному плануванні.

додаток

ПОРАДИ ЩОДО ПОВЕДЕННЯ ТЕСТУ НА КОРОТКОЧАСНУ ПАМ'ЯТЬ

Проілюструємо проведення групового тестування на короточасну пам'ять на базі стандартного кабінету інформатики, що нині є в кожній школі. Для визначення об'єму короточасної пам'яті оберемо метод Джекобса. В [1] є опис тесту на числовому матеріалі, реалізований у вигляді ігрового додатку, з допомогою якого можна перевіряти та тренувати пам'ять. Ми пропонуємо для тестування короточасної пам'яті електронну таблицю (мал. 8 а), б), в)).

Одночасно протягом 5 хвилин можна протестувати 10 учнів. Ще 3-5 хвилин можна відвести для оголошення результатів та їх аналізу. Зміст тестування полягає в тому, щоб протягом 3-5 с запам'ятати стовпчик із семи чисел і протягом 5-6 с їх відтворити по пам'яті. Для роботи з групою рекомендується скористатись програмою для контролю комп'ютерів у мережі зразка

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q		
1	ТЕСТИ НА ШВИДКУ ПАМ'ЯТЬ					ВІДПОВІДІ													
2	№	тест 1	тест 2	тест 3	тест 4	1 відп. аналіз	2 відп. аналіз	3 відп. аналіз	4 відп. аналіз	тест 1									
3	1	4	6	7	5	4	1	6	1	7	1	5	1					4	
4	2	7	7	8	7	7	1	7	1	8	1	7	1					7	
5	3	8	4	4	3	8	1	4	1	4	1	3	1					8	
6	4	5	2	7	9	5	1	2	1	7	1	9	1					5	
7	5	9	9	9	1	9	1	9	1	9	1	1	1					9	
8	6	18	32	76	47	18	2	32	2	76	2	47	2					18	
9	7	235	213	390	438	235	3	213	3	390	3	438	3					235	
10	10 10 10 10																		
11	10																		
12																			
13																			

мал. 8 а)

мал. 8 б)

мал. 8 в)

Суперпам'ять!

М	пам'ять	1	2	3	4	
---	---------	---	---	---	---	--

мал. 8 б)

NetOp School. На робочих місцях учнів (РМУ) та вчителя (РМВ) знаходиться файл MemoMetr.xlsx з електронною таблицею (мал. 8 а)).

Відмінність лише у тому, що файл MemoMetr.xlsx на РМВ має замість листа "Пам'ять" із таблицею для введення чисел (мал. 8 а)) містить ще й листи 1-4 із числами-завданнями для чотирьох тестів (мал. 8 б) та 8 в)). Стовпчики F-M та рядки 10-11(на мал. 8 а) виділено сірим кольором) приховані. У окремих клітинках містяться формули:

B10: =СУММ(G3:G9); C10: =СУММ(I3:I9); D10: =СУММ(K3:K9); E10: =СУММ(M3:M9);

B11: =СРЗНАЧ(B10:E10);

O13: =ЕСЛИ(B11=10;«Суперпам'ять!»;

ЕСЛИ(B11>=8;«Надійна пам'ять!!»;

ЕСЛИ(B11>=7;«Одна дірка в пам'яті є!!!»;

ЕСЛИ(B11>=5;«Дірява пам'ять!!!!»;«Не пам'ять, а решето!!!!»))))).

Звичайно, умови тестування можна змінювати, зокрема, для запам'ятання залежно від віку та розвитку короткочасної пам'яті пропонувати більш складні числа, збільшувати їх кількість, змінювати кількість балів за кожне правильно набране число, нарешті, замість жартівливих оцінок виставляти числові бали. Проте, тоді до подібного тестування не можна підтримувати тривалого й стійкого інтересу. В умовах, коли тестування проводиться для групи, така форма оцінювання підвищує здорову мотивацію учнів. У Щасливському НВК подібні тестування масові, вони проходять періодично, часто у формі змагань між класами. Цим обумовлена назва робочих файлів МемоМетр. Масове тестування почалось у грудні 2013 року, коли було оголошено про створення інтелектуального клубу MEMOMETR (оголошення та умови проведення змагань на мал. 9-10).



мал. 9



мал. 10

Хронометраж сеансу тестування:

- Організаційний момент та інструктаж – 1 хв;
- Відкрити файли МемоМетр.xlsx та перейменувати їх (у кінці додати прізвище учня) – 1 хв;
- Тестування (5с+6с+4с на паузи)*4 = 1 хв;
- Збереження файлів з результатами тестування на РМВ – 2 хв.
- Оголошення та аналіз результатів – 3-5 хв.

IV. ВИСНОВКИ

Не варто ще раз переконувати себе в необхідності замислюватись над сучасним станом якості навчання. Цілком очевидно, що реформувати освіту необхідно і це треба починати знизу, причому, негайно. Верхні щаблі освітньої драбини через свій консерватизм та корумпованість здатні лише заднім числом підхопити досягнення та корисні починання і очолити їх. Але по великому рахунку думати треба про долю підростаючого покоління. А вона за-

лежить в дуже великій мірі від якості освіти. Як відомо, за гроші в Україні можна легко одержати диплом, але скільки відсотків молодих людей з дипломами можуть одержати високооплачуване робоче місце не тільки за кордоном, а й в Україні?

При розгляді методу визначення та застосування коефіцієнту якості навчання, концентрувалась увага виключно на головному суб'єкті навчально-виховного процесу – учневі. Безперечно, що рівень навчальних досягнень учнів залежить у великій мірі від фахової підготовки педагогів та ставлення їх до роботи. Тут є не тільки недоліки, а й провали. Але тут свідомо не розглядалось питання про якість роботи педагогічних працівників, бо це тема іншого дослідження і предмет іншого проекту.

На закінчення слід зауважити, що описаний метод аналізу якості навчання (МАЯК) жодним чином не залежить від виставлених балів у класному журналі, тому вільний від суб'єктивності учителів. Натомість, подібно екзит-полам, що практикуються у виборчих системах, цей метод абсолютно незалежно дає можливість досить точно прогнозувати успішність учнів, а також на основі одержаних результатів здійснювати планування навчальної діяльності й у кінцевому результаті управляти якістю навчання та освіти в цілому. Застосування цього методу може бути динамічною, тому набагато ефективнішою альтернативою так званим електронним журналам, функції яких зводяться лише до статистичної обробки згідно виставлених оцінок.

Використані джерела:

1. http://init2.at.ua/load/navchalni_programi/test_dlja_viznachennja_obsjaqu_korotkochasnoji_pam_39_jati/2-1-0-11

РОЗДІЛ VI.

VI.1 МЕТОДИЧНА РОБОТА УЧИТЕЛЯ ІНФОРМАТИКИ

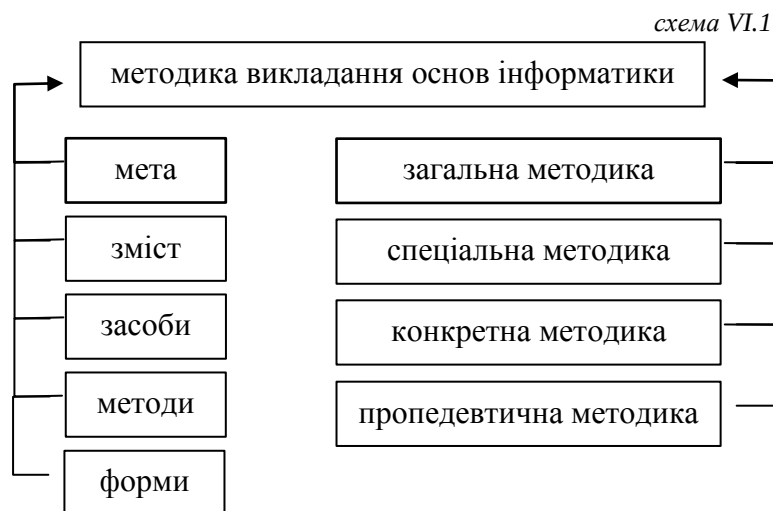
Учительська робота полягає в організації навчання, тому кожен учитель постійно користується спеціальними галузями наукових знань – *методикою* навчання та *дидактикою*.

Методика – це система правил, виклад методів навчання чому-небудь або виконання якоїсь роботи. **Метод** – (від грец. *methodos* – шлях дослідження, теорія, вчення), *спосіб досягнення якої-небудь мети, розв'язання конкретної задачі, сукупність прийомів або операцій практичного чи теоретичного засвоєння (пізнання) дійсності*.

Дидактика (від грец. *didaktikós* – той, хто повчає) – це розділ педагогіки, теорія освіти і навчання. Вона описує закономірності засвоєння знань, умінь та навичок і формування переконань, визначає об'єм і структуру змісту

освіти, удосконалює методи й організаційні форми навчання, виховний вплив навчального процесу на учнів.

Методика викладання основ інформатики розглядає *мету, зміст, засоби, методи і організаційні форми* викладання шкільного курсу інформатики. Слід зауважити, що результати сучасної і класичної дидактики конкретизуються у контексті викладання основ інформатики. І навпаки, проходить осмислення конкретного досвіду, досягнень і проблем практики викладання в школі із позицій загальної дидактики (див. схему VI.1).



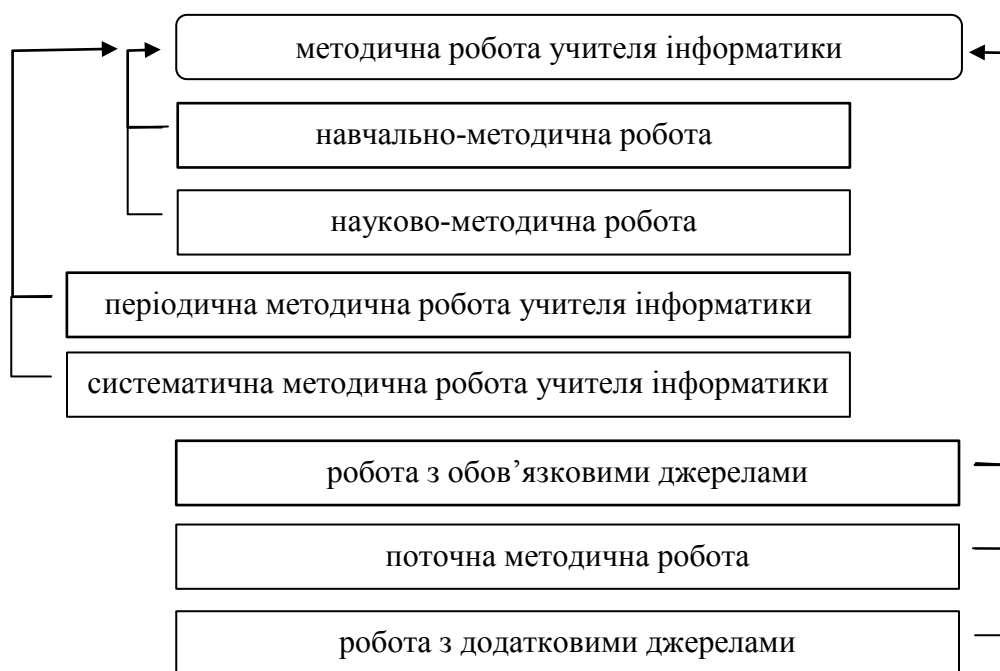
Нажаль, значна частина учителів та викладачів надто спрощено і буквально підходять до використання надбань методичної та дидактичної думки, тобто *ознайомлюються з фаховими періодичними виданнями, сліпо виконують інструкції* щодо календарно-тематичного планування, проведення уроків, обліку знань, умінь та навичок, застосування інтерактивних методів чи комп'ютерних технологій навчання. Але в жодній інструкції немає повних і відповідних конкретним умовам вказівок, як слід поступати у даному випадку. Тому *кожен учитель повинен бути в тій чи іншій мірі методистом*, уміти без інструкцій, творчо й самостійно вирішувати нагальні педагогічні та дидактичні завдання. Іншими словами, *у кожного учителя інформатики повинна бути чітко визначена система методичної роботи*. Вона повинна охоплювати *загальні й спеціальні* аспекти викладання предмету та *конкретну методика*. Необхідність активної методичної роботи учителя основ інформатики зумовлена ще й тим, що, як правило, у школі, районному відділі освіти та районному методичному кабінеті відсутні спеціалісти з методики викладання цього предмету.

Загальні методи викладання – це підходи до проведення практичних занять, застосування інтерактивних технологій навчання, міжпредметних зв'язків тощо. У цьому контексті на заняттях ОШППД ми, наприклад, розглядали концепцію так званого *безпаперового методу проведення уроків інформатики* (БМПУ), тобто застосування через локальну мережу електронних конспектів та сценаріїв уроків (ЕКУ та ЕСУ), електронних дидактичних засобів (ЕДЗ, у тому числі КЕДЗ), електронні форми проведення тестування, практичних і залікових робіт тощо.

Спеціальна методика викладання інформатики передбачає вироблення й застосування підходів до вивчення операційної системи *Windows*, мов програмування, алгоритмів тощо. Ми, зокрема, реалізуємо це у формулюванні конкретних проблем шкільної інформатики та пошуках шляхів їх подолання (згідно планів роботи ОШППД на 2007-2008 та 2008-2009 н.р.), у розробці так званого експрес-курсу алгоритмізації та програмування [6] і т.д.

Конкретна методика викладання основ інформатики складається з окремих питань загальної методики (наприклад, адаптація і розробка програм, планування уроків в окремих класах і з окремих тем), окремих питань спеціальної методики (наприклад, проведення уроків-семінарів, бінарних уроків, розв'язування олімпіадних задач на переборні методи і т.д.).

схема VI.2



Слід розрізняти також методики викладання *пропедевтичних* (підготовчих) курсів ІКТ, комп'ютерної графіки, Web-дизайну тощо та *спеціальні методики*, наприклад, курсу алгоритмізації і програмування, або розв'язування олімпіадних задач з інформатики.

Враховуючи вищесказане, слід погодитись, що кожен учитель основ інформатики у системі власної методичної роботи повинен мати уявлення про її форми, обсяги і періодичність (циклограму).

Форми методичної роботи учителя основ інформатики можуть бути найрізноманітнішими. Вони залежать від рівня *освіти* (дипломований чи атестований), *навантаження* (основний спеціаліст чи сумісник), *досвіду роботи* (категорія, педстаж) тощо. Тому будемо мати на увазі найбільш важливі і загальні (схема VI.2).

1 Методична робота учителя основ інформатики починається з вибору програм і підручників та складання календарно-тематичних планів. Адже, не зважаючи на велику кількість базових програм для універсального і спеціалі-

зованих профілів, факультативних і профільних занять, вибрати ту, яка в повному обсязі відповідає умовам конкретної школи, дуже важко. Враховуючи це, календарно-тематичне планування для кожного класу, універсального чи іншого профілю необхідно забезпечити відповідною пояснювальною запискою.

Значної і кваліфікованої методичної роботи вимагає також адаптація рекомендованих МОН України навчальних підручників до вимог діючих навчальних програм з основ інформатики. Враховуючи те, що підручники [2] місцями не відповідають діючим програмам[1], не зручні у користуванні (нумерація розділів, параграфів і пунктів, обсяг і розпорошеність навчального матеріалу і т.д.) та виходячи з концепції БМПУ бажано розробити серію дидактичних матеріалів у вигляді стислого викладу навчального матеріалу, помістивши їх в окремі папки, наприклад, «Тези до уроків інформатики». Подібним чином можна підготувати серії залікових та тестових завдань чи інші ЕДЗ.

Цей вид методичної роботи можна назвати *роботою з обов'язковими документами та джерелами*.

2 Схожа за формою, але відмінна по змісту є робота з додатковими джерелами. По формі ця робота пов'язана із самоосвітою, але по суті – це одержання нової інформації з обов'язковим впровадженням у практику³². Будь-які додаткові джерела необхідно вивчити, оцінити, узагальнити і адаптувати до конкретних умов (ідеться про узгодженням з психолого-педагогічними та віковими особливостями учнів тощо).

3 Створення саморобних паперових і електронних навчальних документів (дидактичних засобів) – це окремий вид методичної роботи учителя, інформатики у тому числі. Методична складова такого виду роботи полягає у створенні інструкцій використання дидактичних засобів та в плануванні їх застосування. Близькою за змістом і формою є робота над створенням та впровадженням тестових та контролюючих засобів. Для учителя інформатики – це, як правило, електронні засоби. Описаний вид методичної роботи можна назвати *поточною*.

4 Описані вище форми методичної роботи можна об'єднати під загальною назвою – *систематичні*, тобто такі, що виконуються повсякденно, безперервно. Але необхідні також окремі види методичної роботи, які проводяться лише один раз, чи багаторазово, навіть з деяким періодом. Учитель інформатики може написати фахову або методичну статтю, курсову роботу під час навчання на курсах, методичний посібник для учителів, або навчальний посібник для учнів. Кожні п'ять років ми готуємо так звану «папку» для чергової атестації, по сучасному – портфолію, куди «вкладаємо» методичні матеріали різного характеру, змісту і призначення. Це своєрідна збірка власних методичних матеріалів.

³² Слід розрізняти фахову і методичну самоосвіту. Останню можна вважати інтравертивною методичною роботою. Тут же ідеться про екстравертивну методичну роботу, тобто створення та впровадження *методів і організаційних форм* викладання шкільного курсу інформатики.

Класифікацію методичної роботи учителя інформатики можна продовжити, але вона, залежно від педстажу, досвіду та напрямків роботи може бути більш чи менш широкою і глибокою, тому замість єдиного реєстру тут пропонується спрощений орієнтовний перелік.

5 І все ж, враховуючи не стільки обов'язковий, як творчий характер методичної роботи, слід погодитись, що у кожного учителя повинна бути своя близька чи улюблена тема, можливо навіть на рівні своєрідного «хобі». Наприклад, одні учителі більш цікавляться вивченням комп'ютерних мереж, баз даних, Web-дизайну чи комп'ютерної графіки, іншим ближча позакласна робота з обдарованими учнями, олімпіадні задачі, робота в МАН, або об'єктно-орієнтоване програмування. Лише на такому ґрунті можуть виникати серйозні методичні розвідки чи завершені навчально-методичні роботи. Це галузь науково-методичної роботи. У якості прикладів можна згадати теми, розглянуті на заняттях ОШППД:

- Якою бути шкільній інформатиці завтра.
- Письмовий опис алгоритмів як невід'ємна складова розв'язку задач із програмування в школі.
- Метод динамічного програмування у школі.
- Алгоритмічні підходи до розв'язування задач «довгої арифметики».
- Від задачі до задачі.
- КЕДЗ в системі ІТН.
- Роздуми про проблеми шкільних олімпіад з інформатики.

У минулому навчальному році, розглядаючи метод проектів, ми виділяли окремий вид – *індивідуальний проект*. Таким може бути робота над олімпіадною задачею. Після лекції планується проведення практичного заняття по створенню проекту «Олімпіадна задача з інформатики».

Отже, класифікацію методичної роботи можна представити так:

1. Методична робота з обов'язковими документами та джерелами (освітніми стандартами, програмами, підручниками, інструкціями, інструктивно-методичними записками, календарно-тематичне планування тощо);
2. Методична робота з додатковими та спеціальними документами і джерелами (вивчення і впровадження в навчально-виховний процес нових форм і методів навчання та виховання);
3. Поточна методична робота (підготовка поточних методичних та дидактичних засобів навчання);
4. Періодична методична робота (опис і узагальнення власного та вивчення і впровадження запозиченого методичного досвіду);
5. Науково-методична робота.

VI.2 РОБОТА З ДИДАКТИЧНИМИ МАТЕРІАЛАМИ ЯК СКЛADOVA МЕТОДИЧНОЇ РОБОТИ

У пункті II.2 наведено приклад електронного конспекту уроку та розглянуто етапи розробки й використання таких електронних дидактичних засобів (ЕДЗ). Треба зауважити, що набагато ефективніше використовувати серію

ЕКУ, яка охоплює цілу тему. Такі матеріали вище називались комплексними електронними дидактичними засобами (КЕДЗ). Але їх створення вимагає значних затрат часу, тому доцільно об'єднувати процеси створення та використання.

Дійсно, легко помітити що електронний конспект уроку – це своєрідний модуль, який складається з окремих, цілком автономних блоків, записаних у додатках або окремих файлах і поєднаних гіперпосиланнями. Наведемо перелік блоків згаданого вище прикладу:

- а) План уроку (ядро);
- б) Тема, мета, обладнання, література (додаток 1);
- в) Таблиця «Структура програми на PasABC» (додаток 2);
- г) Таблиця «Оголошення і опис величин» (додаток 3);
- д) Таблиця «Команда вводу» (додаток 4);
- е) Таблиця «Команда присвоєння» (додаток 5);
- ж) Таблиця «Команда виводу» (додаток 6);
- з) Вікно програми на PasABC (додаток 7);
- и) Завдання до практичної роботи (додаток 8).

Саме ця обставина дозволяє об'єднувати процеси створення та використання ЕКУ. Поки він не створений, учитель може використати звичайний поурочний план, а перераховані вище блоки застосувати роздільно, як невеликі електронні дидактичні засоби.

Для зразка нижче наведено електронні заготовки до уроку на тему «Операційна система Windows. Об'єкти файлової системи та дерево каталогів. Навігація в пам'яті комп'ютера».

ОПЕРАЦІЙНА СИСТЕМА WINDOWS

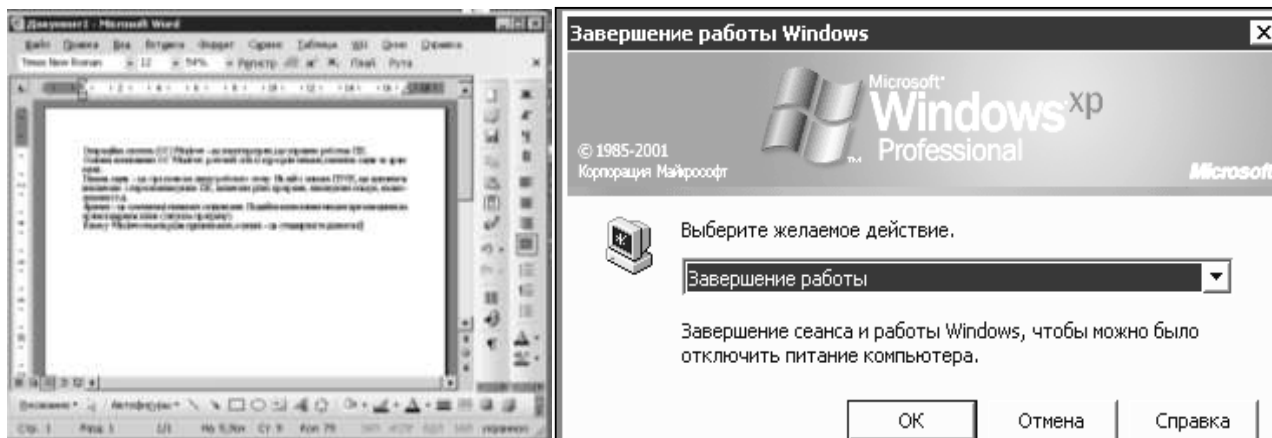
Операційна система (ОС) Windows – це пакет програм, що управляють роботою ПК.

Основні візуальні компоненти ОС Windows: робочий стіл із курсором мишки, панеллю задач та ярликами.

Панель задач – це сіра полоса знизу робочого столу. На ній є кнопка **ПУСК**, що допомагає виключати і перезавантажувати ПК, включати різні програми, виконувати пошук, налагодження і т.д.

Ярлики – це схематичні малюнки з підписами. Подвійне натискання мишки при наведенні на ярлик *відкриває вікно (запускає програму)*.

Вікна у Windows мають різне призначення, основні – це *стандартні та діалогові*.



ОБ'ЄКТИ ФАЙЛОВОЇ СИСТЕМИ ТА ДЕРЕВО КАТАЛОГІВ 10

Файл – це логічно поєднана порція інформації, яка під деяким іменем збережена на носії пам'яті ПК (диску).

Ім'я файлу – це набір допустимих символів.

приклади: Іваненко.doc; Іваненко.bmp.

Чорний Д.doc; Чорний Д.bmp

Windows.exe – виконувані файли, програми.

недопустимі символи: «.»; «.»

Останні символи імені файлу, які починаються «.» називаються розширенням імені файлу. Розширення пишеться латиницею.

Каталог (папка, директорій) – об'єкт, збережений в пам'яті ПК під деяким іменем, і містить папки (підкаталоги) та файли.

Каталог, що містить всередині інші каталоги називається старшим каталогом, найстарший каталог на диску чи в пристрої називається кореневим каталогом і позначається «\».

приклади: D: \ klass \ 7_a \ Group_1.

Папка Group_1 є підкаталогом папки 7_a, яка в свою чергу міститься в папці klass, що є підкаталогом кореневого каталогу.

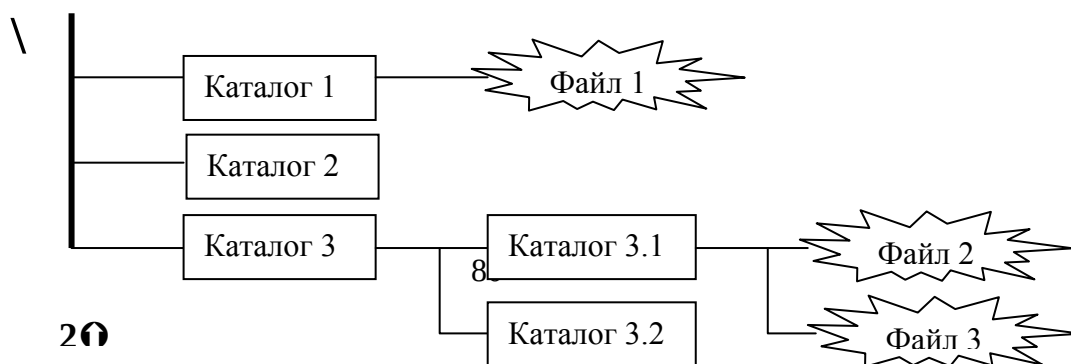
Дерево каталогів 20 – це система каталогів на диску.

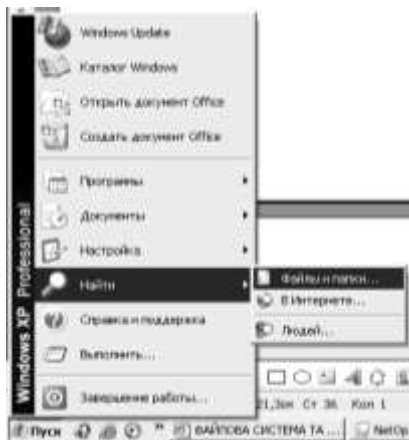
НАВІГАЦІЯ В ПАМ'ЯТІ КОМП'ЮТЕРА (ПК) 30

Під навігацією розуміють «подорож» по пам'яті комп'ютера, тобто переходи з диска на диск, відкриття папок та вихід з них, створення, копіювання і вставка файлів чи папок, а також пошук 40

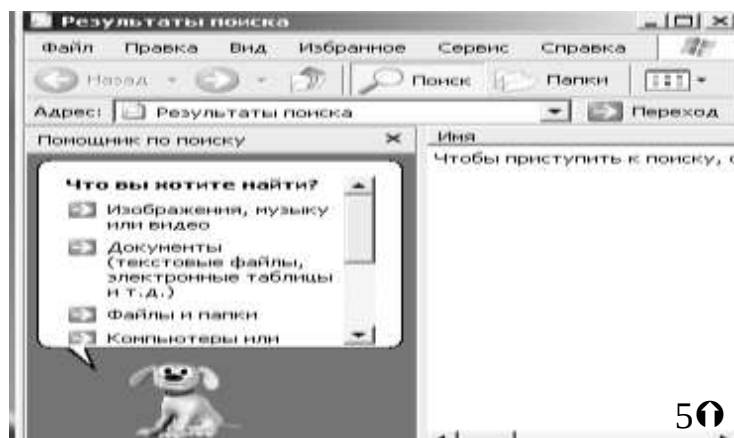
Пошук файлів і папок. Щоб знайти потрібні файл чи папку, треба виконати команду НАЙТИ і інші дії у вікні «Результати пошука» 50. Команду НАЙТИ можна виконати з допомогою панелі задач, або з допомогою ПРОВІДНИКА 60.

ПРОВІДНИК 70 – це вікно, яке практично співпадає з вікном МОЙ КОМП'ЮТЕР, але зліва має панель «Папки» 80, що відображає дерево каталогів активного диска чи пристрою.





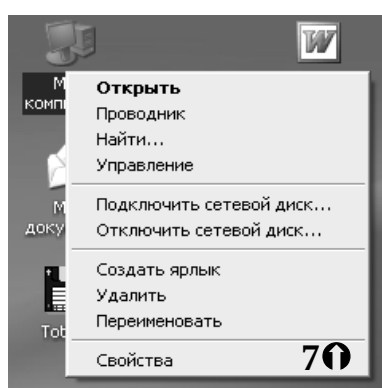
40



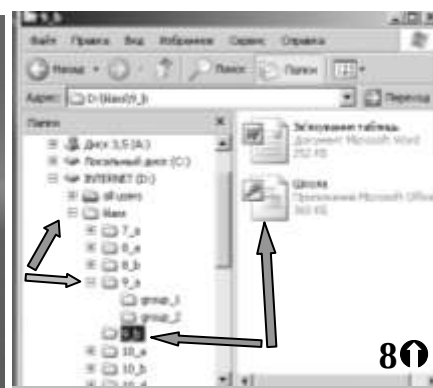
50



60



70



80

Домашнє завдання: § 7, § 9-10, повт. § 6

ПИТАННЯ ДО ЗАЛІКУ:

1. Файлова система та дерево каталогів: 10
 - Поняття файлу. Ім'я та розширення імені файлу. Шлях до файлу. Повне ім'я файлу. Виконувані файли;
 - Дерево каталогів. Кореневий каталог. Підкаталоги;
2. Створення, копіювання, переміщення, перейменування та стирання файлів і каталогів. Вхід в каталог і вихід з нього. Відкриття та закриття файлів.
3. Навігація в пам'яті ПК. Провідник:
 - Пошук файлів і папок;
 - Знаки «*» та «?» в іменах файлів і папок; 30

Ці заготовки електронних дидактичних засобів займають лише дві сторінки, містять великий обсяг навчального матеріалу. Для зручності користування вони зв'язані тимчасовими гіперпосиланнями. Ними можна вигідно замінити підручник, адже подібна тема у підручнику, як правило, займає у 5-10 разів більше сторінок. З часом такі заготовки можуть бути матеріалом для конструювання ЕКУ або КЕДЗ.

VI.3 ОСОБЛИВОСТІ ВИКЛАДАННЯ ОСНОВ ІНФОРМАТИКИ В 9-11 КЛАСАХ

2009-10 н.р. особливий тим, що вперше введено навчальний предмет «Інформатика» в дев'яті класи. Це пов'язано не тільки з переходом на 12-річне навчання, а й із зміною концепції вивчення ІКТ в загальноосвітній школі. У зв'язку з цим введена нова програма [2] та кілька альтернативних підручників [18], [19], [20]. Поверховий погляд на ці джерела не виявляє нічого особливого: ті ж вступні теми, традиційний порядок вивчення інформатики на першому році вивчення інформатики, приблизно такий же розподіл навчальних годин. Правда, підсилено присвячений операційній системі розділ, збільшено обсяг навчального матеріалу по темі «Комп'ютерні мережі», широко представлено тему «Комп'ютерна графіка», яку до цього вивчали в 10-11 класах оглядово.

Протягом 2010-11 та 2011-12 н.р. вслід за дев'ятьма класами були введені нові програми та підручники з інформатики й для 10-11 класів. Для них прийнято правильним буде все, що нижче написано стосовно викладання інформатики в 9-х класах.

Але ще при першому, загальному, знайомстві з новим курсом виникає цілий ряд суттєвих питань. Ознайомлення з критеріями оцінювання навчальних досягнень учнів викликає сумнів, що за діючою програмою, при встановленій кількості навчальних годин та з допомогою рекомендованих МОН України підручників можливо досягти високого рівня навчальних досягнень, адже він передбачає, щоб знання, вміння і навички учня відповідали вимогам державної програми **у повному обсязі**.

Щоб переконатись у цьому, досить виписати всі поняття та їхні особливості, які відносяться до окремо вибраної теми та зіставити їх із кількістю відведених програмою навчальних годин. Крім того слід враховувати вікові особливості учнів 9 класу, їх уміння визначати й опрацьовувати навчальну, та сприймати наукову інформацію. Для прикладу візьмемо тему «Поняття файлу, каталогу. Ім'я файлу та каталогу, розширення імені файлу. Імена зовнішніх запам'ятовуючих пристроїв, шлях до файлу», яка є частиною теми «Системне програмне забезпечення» (додаток 1). Протягом одного уроку необхідно вивчити та закріпити достатньо абстрактні поняття (див. програму [2]):

- файлу;
- імені файлу;
- розширення імені файлу;
- повного імені файлу;
- каталогу;
- кореневого каталогу;
- підкаталогу;
- старшого каталогу;
- дерева каталогів;
- зовнішнього запам'ятовуючого пристрою.

Особливістю вивчення основ інформатики в умовах практичної форми проведення навчальних занять є необхідність унаочнити перераховані понят-

тя з допомогою комп'ютера. В умовах групових занять це вимагає відведення 3-5 хв. на кожне з них, тобто всього 30-45 хв.

У пропонуваному календарно-тематичному плані з інформатики для 9 класу відповідно до діючої програми вивчення названої теми включено в урок № 6 «Загальні відомості про системне, службове та прикладне програмне забезпечення. Класифікація, основні функції та складові операційних систем. Поняття про ядро операційної системи, інтерфейс користувача, драйвери та утиліти. Різновиди інтерфейсу користувача. *Поняття файлової системи, відмінності між поширеними файловими системами. Поняття файлу, каталогу. Ім'я файлу та каталогу, розширення імені файлу. Імена зовнішніх запам'ятовуючих пристроїв, шлях до файлу*», отже, якщо враховувати важливість і ємкість понять першої частини теми «Загальні відомості про системне, службове та прикладне програмне забезпечення. Класифікація, основні функції та складові операційних систем. Поняття про ядро операційної системи, інтерфейс користувача, драйвери та утиліти. Різновиди інтерфейсу користувача» на вивчення другої частини теми можна відвести не більше 50% часу уроку, тобто до 25 хв. Проведений аналіз переконує, що *вивчення вищеназваної теми в умовах плану і програми можливе лише в оглядовому вигляді*, що недопустимо. Відомо, що освіта – це те, що залишається після навчання. Можливо однією з причин низького рівня сучасної освіти є перенасиченість програм та підручників з одного боку та недостатня кількість відведених програмою навчальних годин з іншого? Якщо це так, то, мабуть, варто звужити програмний матеріал і вивчати лише те, що обов'язково повинно залишитись після навчання.

У діючій програмі існує, на наш погляд, чимало «вузьких місць». Звернемо увагу на відсутність чіткої послідовності вивчення понять та вироблення умінь і навичок. Щоб побачити їх, досить ще раз подивимось на тему «Системне програмне забезпечення», на вивчення якої відводиться 7 годин. Тема включає три практичні роботи³³. "Розкиданість" навчального матеріалу теми видно, як кажуть, неозброєним оком.

МОН України рекомендує для вивчення інформатики в 9 класі 4 підручники різних авторських колективів. Наявність широкого вибору підручників безперечно є позитивним фактом. Але слід погодитись, що кожен авторський колектив має право на власну інтерпретацію курсу, а в умовах існування недоліків програми, про що сказано вище, різні підручники в більшій чи меншій мірі, але обов'язково спотворюють курс у цілому. Наприклад, у підручнику [18] поняття інтерфейсу користувача ОС Windows та об'єктів файлової системи описано коротко й чітко:

Інтерфейс користувача – сукупність засобів, які забезпечують обмін даними між користувачем і ОС.

Об'єкти файлової системи: папки, файли, ярлики.

Файлова система – структура збереження даних на зовнішніх носіях і сукупність програм, які забезпечують роботу з цією структурою. Як правило,

³³ Для зручності перша практична робота та відповідний їй матеріал виділено підкресленим курсивом, друга практична робота з відповідним матеріалом виділено курсивом.

операційна система може працювати з кількома файловими системами ([18], стор.77).

У [20] ці ж поняття подано поширено й описово:

Інтерфейс (діалог, обмін інформацією) між користувачем та комп'ютером.

WIMP-інтерфейс (*Window* – вікно, *Image* – образ, *Menu* – меню, *Pointer* – указівник), або **графічний інтерфейс**.

Використовуючи цей вид інтерфейсу, команди комп'ютеру надають за допомогою графічних образів (меню, вікон та інших елементів). Саме цей вид інтерфейсу зrealізовано в ОС *Windows*.

Об'єкти файлової системи перераховуються так: папки, файли і інші об'єкти.

Файлова система – частина будь-якої операційної системи, що відповідає за організацію зберігання та доступу до інформації на будь-яких її носіях ([20], стор. 129).

Спроба оптимального підходу до опису інтерфейсу користувача у [19] (п.5.6, стор. 54): вступне пояснення, означення, графічна ілюстрація, детальний опис. Проте, чіткості, як у [18], чи повноти, як у [20] немає. Ось означення:

Інтерфейс користувача – це сукупність правил і засобів, застосовуючи які користувач може розв'язувати задачі за допомогою комп'ютера.

Слово "правил" підкреслено, адже воно лише заплутує учня, на наш погляд, досить обійтись терміном «засобів», як у [18]. Завдання інтерфейсу користувача – забезпечувати обмін інформацією між користувачем та ОС, а не розплівчате «розв'язування задач», як у [19].

Ілюстрація поняття «командний рядок» (мал.5.4) відноситься до *MS DOS*, тому неактуальна, а «графічний інтерфейс» описано формально і не наочно.

У п. 6.4 перераховано та описано об'єкти операційної системи: робочий стіл, програми, папки, документи, ярлики, панель завдань, кнопка ПУСК, головне меню системи, кошик та вказівник мишки. Але поняття об'єкти файлової системи, передбачене програмою у темі 3 «Системне програмне забезпечення», у [19] розглядається п.17 розділу 5.

У розділі VI (Методична робота учителя інформатики) наведено приклад заготовки для електронного конспекту уроку «Операційна система *Windows*. Об'єкти файлової системи та дерево каталогів. Навігація в пам'яті комп'ютера», де зовсім відсутній опис файлової системи, а мова йдеться лише про об'єкти файлової системи, зокрема про файли та каталоги. На наш погляд, для учнів 9 класу такої інформації цілком досить, адже вони все одно не зможуть повністю усвідомити терміни зразка FAT32 та NTFS.

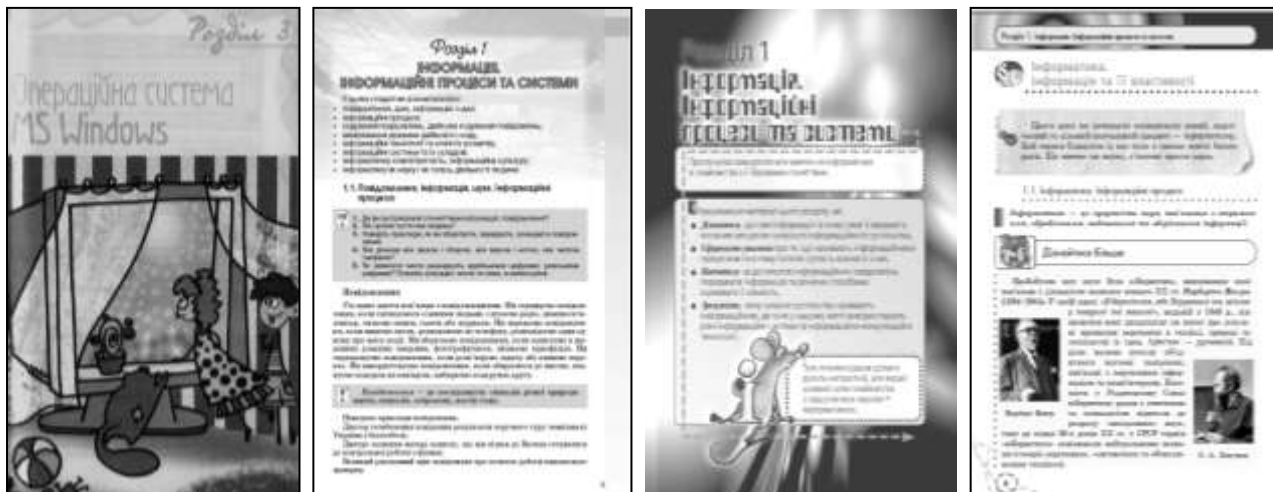
Тут ми розглядаємо питання конкретної методики, але без урахування загальних методологічних засад заглибитись у конкретику не можливо. Прикро, але авторські колективи нових підручників змушені не тільки балансувати між своїм баченням предмета та вимогами державних стандартів, освітніх концепцій і програм, а ще часто обмежені умовами участі в конкурсі під-

ручників, необхідністю виграти тендер тощо. Про це свідчать критерії, згідно яких, наприклад, давалась оцінка експертної комісії по рецензуванню підручників з інформатики для 11 класу в грудні 2010 р. Ці критерії були непродумані й формальні, серед обов'язкових вимог до підручників з інформатики чільне місце займали «Мова викладу навчального матеріалу в рукописі», «Реалізація у змісті рукопису підручника виховних можливостей навчального предмета», «Українознавче наповнення змісту рукопису підручника», в т.ч. й при описі інтерфейсів, «відображення історії України» тощо. Найбільш відповідають цим вимогам підручники авторського колективу Н.В. Морзе та ін. Але від цього підручники не стали кращими.

Підручник – це не засіб колективного навчання, він призначений для допомоги при індивідуальному опануванні навчальним матеріалом, навіть якщо клас використовує підручник на уроці під керівництвом учителя. Тому він не повинен бути переобтяженим, наприклад, елементами інтерактивного навчання, які належать до методики і можуть бути присутніми у методичному посібнику для учителя. Чи знизилась би якість підручника [19], якби завдання 1: «Наведіть приклади інформаційних систем. Складіть список п'яти інформаційних систем» було не в рубриці «Працюємо в парах» (стор. 32)? Те ж питання можна поставити щодо завдання 1 рубрики «Працюємо самостійно»: «Поясніть кожний із термінів, які використовують для тлумачення комп'ютера як програмно-керованого пристрою, призначеного для автоматизованого опрацювання даних» (там же). У підручнику повинні бути питання та завдання, призначені для учнів. Їх можна поділити на простіші та складніші. Але на цьому й закінчується функція авторів, як використати питання та завдання нехай «болить голова» викладача, адже це вже методика. Автори були б «такими ласкавими», якби до свого підручника додали методичний посібник. Звідси напрашується закономірний висновок: замість кілької альтернативних підручників МОН України слід фінансувати один підручник, і обов'язковий до нього методичний посібник. Якщо знайдеться інший авторський колектив, який запропонує альтернативу, нехай приймає участь у конкурсі за власний рахунок і буде винагороджений, коли виграє його. Підручник виданий державним коштом повинен бути еталоном.

Незаперечна істина, що підручником користуватись тим легше, чим він стисліший. Крім того, неможливо спростувати відому істину «Стислість – сестра таланту». Але чи можна назвати стислими діючі підручники з інформатики для 9 класу, які всього на 35 уроків містять по кілька сотень сторінок (наприклад, [19] – 344 сторінки)? Ще гостріше питання: чи легко *вперше* викладати інформатику в 9 класі при наявності недосконалої програми, чотирьох громіздких підручників без методичних посібників? Відповідь очевидна, бо питання риторичне.

Поряд із стислістю для ліцензованого підручника не менш важливими є науковість, коректність та логічна послідовність. Якщо уважно переглянути переглянути згадані підручники, можна знайти достатньо прикладів порушення цих принципів. Правда, через розпорошення і перемішаність матеріалу на сторінках підручників не завжди можна чітко відрізнити ці порушення від придірок. Але у подібних ситуаціях більшість сумнівів читача слід заархивувати у негатив авторам. Щоб уникнути голослівності, пропонуємо приклад. Мова йдеться про інформаційну систему. З одного боку вона має дві обов'язкові складові: апаратну та програмну, з іншого боку, в підручнику [19], п.33 на стор.29 стверджується, що сучасний комп'ютер є інформацій-



ною системою. Якщо й погодитись на це, то лише за умови, що до складу комп'ютера, крім апаратури, обов'язково входить програмне забезпечення. Але чи слід дев'ятикласників навантажувати такими небезсумнівними і складними деталями?

На попередніх сторінках здебільшого ми аналізували діючі підручники. Нижче проведено порівняльний аналіз трьох із них: Й.Я. Ривкінда та ін. [18], Н.В. Морзе та ін.[19] та І.Л. Володіної, В.В. Володіна [20]. Всі вони мають у більшій чи меншій мірі загальні вади сучасних підручників. Це і неякісний папір та поліграфія, і громіздкість та переобтяженість зайвою деталізацією, і необгрунтована данина гуманізації та демократизації навчання, що виражається у зловживанні часто недоцільними ілюстраціями та унаочненнями, і науковий рівень, що не повністю відповідає віковим особливостям учнів 9 класів.

Кожен зайвий фон для тексту, обрамлення або оформлення сторінки чи завиток в нумерації, як проілюстровано вище, розсіює увагу, відволікає від основного, збільшує психологічне навантаження. Сучасні учні не дуже люблять читати та самостійно працювати з книгою, тому всілякі рубрики зразка «Ви дізнаєтесь», «Вивчаємо», «Діємо», «Працюємо в парах», «Поглиблюємо знання», «Обговорюємо», «Дізнайтеся більше», «Перевірте себе» тощо, особливо в умовах цейтноту відведеного програмою часу можуть викликати не сумісні із зацікавленням емоції та мотивацію.

Порівняння підручників проведемо на прикладі вже згаданої теми «Поняття файлу, каталогу. Ім'я файлу та каталогу, розширення імені файлу. Імена зовнішніх запам'ятовуючих пристроїв, шлях до файлу».

підручник	[18]	[19]	[20]
кількість сторінок	11,5	12	10
додаткові поняття ³⁴	12	3	3
відсутні описи ³⁵	3	-	-
додаткові описи	7	4	4
запитання	18	10	13
завдання	9 (з підпунктами 40)	13	19 (в т.ч. 7 додаткових)
практичні роботи	практична (17 п.)	лабораторна (16 п.)	практична (12 п.)

Звідси випливають висновки:

- 1) Кількість сторінок, відведених для цієї теми приблизно однакова, але в усіх підручниках занадто велика для одного уроку.
- 2) В усіх підручниках є додаткові, не передбачені програмою поняття. Але у першому їх значно більше (12), ніж у інших (по 3), тому підручник [18] у цьому відношенні гірший.
- 3) У підручнику [18] відсутній опис трьох понять, передбачених програмою, що також знижує його якість.
- 4) Всі підручники містять додатковими описами (перший – 7, решта – по 4).
- 5) У [20] введені поняття *системних* (службових, створених операційною системою) та *користувацьких* (створених користувачем) **папок**, причому *Робочий стіл* названо головною системною папкою, що містить усі інші папки комп'ютера. Таким чином допущено щонайменше три некоректності. По-перше, Робочий стіл містить *ярлики*, серед яких можуть бути і позначення папок. Ярлики ж – це кнопки запуску додатків та відкриття вікон, у тому числі й папок, файлів чи пристроїв. По-друге, прийнято вважати, що папку можна відкрити, а також вийти з неї. Як відомо, з Робочого столу виходу немає. По-третє, Робочий стіл більш коректно співставляти не з папкою, а з вікном *Windows*, хоча і тут немає повної аналогії.
- 6) Зустрічаються й інші, не зовсім прозорі або не коректні описи. Так у тому ж [20] дано означення поняття формату файлу, перед яким описано назву файлу (стор. 124). Якщо термін «формат» розуміти, як загальний вигляд або запис чогось, то слід розрізняти два поняття «формат імені файлу» та «формат файлу», причому, тут важливіше перше.
- 7) Виходячи з практичної форми вивчення курсу інформатики, практичним роботам слід відводити провідну роль, а рекомендовані підручники повинні це забезпечувати. Підручники [18], [20] містять всі 12 практичних робіт, передбачених програмою. Цілком природно, щоб вони повністю відповідали пунктам навчальних досягнень: *розпізнає, класифікує, вміє та використовує*. Але у програмі важко чітко розділити частини цих пунктів,

³⁴ Додаткові - це не передбачені програмою поняття та описи (NTFS, CDFS, користувацька папка, папка першого рівня, порожній файл, вигляд папок у Windows Vista, відмінності позначень у Windows XP та Windows Vista, відтворення музичних файлів та відеозаписів і т.д.)

³⁵ Старший каталог, підкаталог, дерево каталогів і т.д.

що відносяться до різних практичних робіт. У підручник [19] замість практичних робіт включено лабораторні роботи.

На практичних та лабораторних роботах слід зупинитись окремо. Звернімо увагу на оптимальну кількість та на форму і зміст кожної. Якщо їх вважати заліковими, то передбачена програмою кількість занадто велика (12 робіт на 35 уроків, тобто на кожному третьому-четвертому уроці). Але в програмі не вказано, що всі практичні роботи обов'язкові, тому їх можна поділити на залікові та поточні. У [19] кількість лабораторних робіт вісім, тому усі можна вважати заліковими.

Для заліку форма та зміст лабораторних робіт у [19] безперечно кращі, ніж у решти підручників. Вони оформлені у вигляді таблиці, що містить конкретні завдання, виконання яких легко фіксувати в умовах одночасного контролю групи учнів. Оцінювання можна зробити як під час роботи, так і після її закінчення, адже для кожного питання вказано кількість балів.

Цим перелік зразків відступу від програми, порушення науковості, логічної послідовності та строгості в названих підручниках не обмежується, але очевидно, що у такій ситуації необхідно знайти оптимальний вихід і відповідну методику викладання. Це можливо, наприклад, шляхом стислого викладу матеріалу, подібно експрес-курсу з теми «Алгоритмізація та програмування», представленого засобами електронних дидактичних засобів навчання, зокрема у вигляді електронних сценаріїв уроків (ЕСУ), електронних конспектів уроків (ЕКУ), електронних роздаткових матеріалів (ЕРМ) та електронних тестуючих систем (ЕТС).

Окремої уваги потребує складання календарно-тематичного плану з інформатики для 9 класу. Є необхідність переміщення тем та перерозподілу навчальних годин. Тема кожного уроку потребує стиснення формулювання (додаток 2).

Розуміючи, що змістові зв'язки між багатьма темами курсу є достатньо слабкими, автори у пояснювальній записці зауважують, що вони не мали на меті догматизувати послідовність викладання матеріалу, тобто стандартизація певного порядку їх вивчення є недоцільною. Вчителі інформатики *«можуть змінювати порядок вивчення і обсяг тем курсу залежно від рівня підготовки учнів і технічного оснащення школи, вибудовуючи в такий спосіб найбільш доречну для конкретного навчального закладу або класу траєкторію навчання»*, але при умові, що змінюючи порядок тем курсу, вчитель не порушуватиме *«порядок викладання тем, між якими є суттєві змістові залежності»* [2].

У програмі [2] зазначено, що *«кількість тематичних оцінок встановлюється учителем, програма дозволяє об'єднувати теми, на вивчення яких відведено 2-4 навчальні години. Зокрема доцільно поєднати оцінювання з тем 1 і 2, 3 і 4, 6 і 7. Форму проведення тематичних оцінювань учитель обирає самостійно: контрольні роботи, тестування, комплексні практичні роботи, захист навчальних проектів тощо»*.

Враховуючи різні підходи до вивчення важливих тем курсу у різних підручниках, при складанні календарно-тематичного плану обов'язкова

прив'язка до вибраного підручника. Порівняльний аналіз трьох підручників [18], [19] та [20], на наш погляд, надає перевагу підручнику авторського колективу Н.В. Морзе, В.П. Вембер та О.Г. Кузьмінської. Найсуттєвішими аргументами на його користь ми бачимо підходи до вивчення теми «Комп'ютерна графіка», «Службове програмне забезпечення» та «Основи інтернету». Скориставшись порадою у пояснювальній записці [2]: *«якщо у переліку вказано кілька програм певного типу, то це означає, що можна використовувати будь-яку з них, на вибір учителя»*, автори реалізують ідею вивчення векторної графіки на базі вмонтованих графічних можливостей *Microsoft Office*, зокрема у *Microsoft Word* та *Microsoft Power Point*. Беручи до уваги вікові особливості учнів 9 класу, це дійсно простіше і доцільніше, ніж розгляд векторної графіки в *CorelDraw*, яке, як вищий ступінь, можна вивчити в наступні роки. Враховуючи те, що упаковування й архівація інформації та антивірусна профілактика у наш час міцно пов'язані з роботою в Інтернеті, природно поєднати теми «Службове програмне забезпечення» та «Основи інтернету». Але першість за найвищий науковий рівень, найчіткішу структуру викладу слід віддати підручнику авторського колективу під керівництвом Й.Я. Ривкінда [18].

При складанні календарно-тематичного плану (додаток 2) тут зближено також теми «Системне програмне забезпечення» та «Локальні мережі». Незначний, але потрібний перерозподіл навчального часу зроблено також на користь теми «Апаратне забезпечення інформаційних систем» за рахунок резерву. Це дозволить детальніше познайомити учнів з роботою пристроїв вводу інформації, адже потреба управляти комп'ютером, в тому числі і через клавіатуру, виникає значно раніше за вивчення текстового редактора.

Рамки матеріалу дуже обмежують методичне дослідження викладання інформатики в 9 класі, тому тут піднято лише найбільш загальні та суттєві питання, подальша робота в цьому напрямку може привести до створення експрес-курсу інформатики для 9 класу та методичного посібника до нього.

додаток 1

Фрагмент програми з інформатики для 9 класу
(тема "Системне програмне забезпечення", 7 годин)

Загальні відомості про системне, службове та прикладне програмне забезпечення. Класифікація, основні функції та складові операційних систем. Поняття про ядро операційної системи, <u>інтерфейс користувача</u>, драйвери та утиліти. <u>Різновиди інтерфейсу користувача</u>. <u>Поняття файлової системи, відмінності між поширеними файловими системами</u>. <u>Поняття файлу, каталогу</u>. <u>Ім'я</u>	Учень пояснює: відмінність між системним, службовим та прикладним програмним забезпеченням; поняття ядра операційної системи, <u>інтерфейсу користувача</u>, драйвера та утиліти; <u>поняття файлової системи</u>; <u>відмінності між поширеними файловими системами</u>; <u>зміст шляху до файлу</u>; <u>поняття файлу та каталогу</u>; <u>поняття типу файлу</u>; <u>поняття та призначення ярликів</u>; необхідність періодичної перевірки та очищення дисків; Учень описує: призначення та основні функції операційної системи; <u>основні правила роботи з об'єктами файлової системи</u>; <u>різновиди інтерфейсу користувача</u>;
--	--

файлу та каталогу, розширення імені файлу. Імена зовнішніх запам'ятовуючих пристроїв, шлях до файлу. <u>Робота з основними елементами графічного інтерфейсу користувача операційної системи. Використання вікон, меню, елементів керування.</u> Робота з об'єктами файлової системи: створення, копіювання, перейменування, переміщення та видалення об'єктів. Використання ярликів. Використання буфера обміну. Пошук інформації на комп'ютері. Запуск на виконання програм. Типи файлів. Зв'язок типів файлів з програмами та з розширеннями імен файлів. Використання автономної та онлайнної довідки операційної системи. Встановлення й видалення програм. Відновлення видалених даних. Програма перевірки й очищення дисків. Дефрагментація дисків. Контрольні точки відновлення операційної системи. <u>Практична робота №2. Робота з інтерфейсом користувача операційної системи.</u> <u>Практична робота №3. Робота з об'єктами файлової системи.</u> <u>Практична робота №4. Пошук інформації на комп'ютері.</u>	способи запуску програм на виконання; спосіб відновлення видалених даних; методику встановлення й видалення програмного забезпечення; Учень розпізнає: файли та каталоги; імена, розширення імен та типи файлів; файли, яким зіставлені програми; стандартні імена зовнішніх запам'ятовуючих пристроїв комп'ютера; Учень класифікує: <u>операційні системи за типом інтерфейсу користувача, за кількістю користувачів та програм, що працюють у системі одночасно;</u> Учень вміє: <u>переміщувати, відкривати, розгортати, згорнути вікна та змінювати їхній розмір;</u> визначати й записувати шлях до файлу; переходити до файлу за заданим шляхом; виділяти об'єкти та групи об'єктів для виконання операцій над ними; створювати каталоги; створювати ярлики (посилання на файли, каталоги або диски); перейменовувати файли та каталоги; видаляти файли та каталоги; копіювати й переміщувати файли та каталоги з використанням та без використання буфера обміну; запускати на виконання програми; відкривати файли, типи яких зв'язані з програмами; звертатися до служби технічної підтримки виробників операційної системи та прикладного програмного забезпечення; встановлювати й видаляти програми за допомогою спеціальних засобів, що надаються операційною системою; відновлювати видалені файли та папки; створювати резервні копії файлів та папок; створювати контрольні точки відновлення та визначати розклад їх автоматичного створення; повертати стан системних файлів до контрольної точки відновлення; знаходити на комп'ютері необхідну інформацію в автоматизованому режимі; визначати необхідність дефрагментації дисків; Учень використовує: <u>елементи керування для виконання дій в середовищі операційної системи;</u> меню вікна папки та головне меню операційної системи; буфер обміну для копіювання та переміщення файлів, каталогів та ярликів; автономну та онлайнну довідку операційної системи; засоби автоматизованого пошуку інформації на комп'ютері; засоби відновлення стану системних даних; програму перевірки й очищення дисків; програму дефрагментації дисків.
--	--

додаток 2

Календарно-тематичне планування курсу інформатики для 9 класу

I семестр

1. Інформація. Інформаційні процеси та системи (2 год.)	
1	Інформаційні процеси. Інформація, її різновиди, подання, представлення та вимірювання.
2	Інформаційні системи та технології, їх розвиток і застосування. Апаратне та програмне забезпечення інформаційної системи. Інформаційна культура та інформатична компетентність.
2. Апаратне забезпечення інформаційних систем (5 год.)	

3	Архітектура персонального комп'ютера. Призначення пристроїв введення, виведення, зберігання та обробки інформації. Робота з ручним маніпулятором mouse.
4	Процесор та запам'ятовуючі пристрої. Периферійні пристрої. Розвиток обчислювальної техніки. Правила техніки безпеки під час роботи на комп'ютері.
5	Клавіатура. Робота з клавіатурним тренажером. Введення тексту.
6	Редагування тексту з допомогою <i>Notepad</i> (блокнота).
7	<i>Практична робота №1</i> . Залік.

Тематичне оцінювання

3. Системне програмне забезпечення (7 год.)	
8	Системне, службове та прикладне програмне забезпечення. Операційні системи. Диски, файли та каталоги.
9	Графічний інтерфейс користувача операційної системи. Вікна, меню, елементи керування. Робота з об'єктами файлової системи.
10	<i>Практична робота №2</i> . Робота з інтерфейсом користувача операційної системи.
11	<i>Практична робота №3</i> . Робота з об'єктами файлової системи.
12	Пошук інформації в комп'ютері. Довідкова система. Виконання програм. Типи файлів та їх зв'язок з програмами.
13	Інсталяція програм та відновлення даних. Сервісні програми.
14	<i>Практична робота №4</i> . Пошук інформації на комп'ютері. Залік.

Тематичне оцінювання

4 Комп'ютерні мережі. Локальні комп'ютерні мережі (3 год.)	
15	Глобальні та локальні комп'ютерні мережі, їх апаратне й програмне забезпечення. Поняття про сервери та клієнти. Мережні протоколи.
16	Доступ до ресурсів локальної мережі, вхід та навігація у ній. Спільне використання файлів, папок та пристроїв.
17	Віддалене керування комп'ютером. <i>Практична робота №7</i> . Спільне використання ресурсів локальної мережі. Залік.

Тематичне та семестрове оцінювання

II семестр

5. Службове програмне забезпечення (3 год.)	
18	Комп'ютерні віруси та їх класифікація. Антивірусні програми та антивірусна політика. <i>Практична робота №5</i> . Захист комп'ютера від вірусів.
19	Стиснення інформації. Програми-архіватори. <i>Практична робота №6</i> . Архівування та розархівування даних.
20	Форматування та копіювання дисків. Запис інформації на оптичні носії. Залік.
6. Основи інтернету. Пошук в інтернеті. (3 год.)	
21	Глобальної мережі та інтернет. Адресація в інтернеті. Функції провайдера та послуги інтернету. Гіпертекстові документи. Навігація в інтернет.
22	Браузери та їх налаштування. Вибір системи кодування. Веб-сторінки, веб-сайти та їх збереження на локальному комп'ютері.
23	Засоби пошуку інформації в Інтернеті. Принципи функціонування веб-каталогів та пошукових систем. <i>Практична робота № 8</i> . Пошук інформації в Інтернеті. Залік.

Тематичне оцінювання

6. Основи роботи з текстовою інформацією. (5 год.)	
24	Системи обробки текстів та середовище текстового редактора. Формати текстових файлів. Робота з текстовими документами.
25	Введення й редагування тексту. Операції з фрагментами тексту. <i>Практична робота №9</i> . Робота з текстовими фрагментами.
26	Послуги текстового редактора. Форматування шрифтів і абзаців. <i>Практична робота №10</i> . Введення, редагування й форматування тексту.
27	Об'єкти в текстовому документі. Написи, заголовки та формули. Залік.

Тематичне оцінювання

7. Графічні зображення. Основи векторної графіки. (3 год.)	
28	Графічні можливості офісних програм. Панель “Малювання”. Векторні зображення, їх побудова та обробка.
29	Інструменти малювання. Об’єкти та операції з ними. Автофігури. Параметри об’єктів.
30	Взаємозв’язок між текстовими та графічними компонентами у текстових документах. <i>Практична робота №12.</i> Створення векторних зображень. Залік.
7. Комп’ютерна графіка та її види. Растрова графіка (4 год.)	
31	Комп’ютерна графіка та засоби обробки графічних даних. Растрові й векторні зображення та їх властивості. Графічні формати. Робота з графічними файлами.
32	Панель інструментів Paint. Редагування малюнків та створення надписів.
33	Використання головного меню та палітри кольорів. Роздільна здатність, глибина кольору та якість растрових зображень.
34	<i>Практична робота №11.</i> Створення растрових зображень. Залік.
Тематичне та семестрове оцінювання	
Резерв навчального часу. (1 год.)	
35	Підсумки за навчальний рік.

На закінчення слід ще раз звернути увагу на методичну та дидактичну підтримку підручників. Пройшло три роки, але методичних посібників до нових підручників з інформатики так і не з’явилося. Дидактичні матеріали є, але їх розповсюдження платне, видання та продаж залежить від комерційних структур, тому малодоступне. В квітні 2012 р. автору, як члену комісії з встановлення грифів МОН України навчально-методичної літератури з інформатики довелось рецензувати комплексний зошит для контролю знань з інформатики для 10 класу відповідно академічного рівня. Із зрозумілих причин не називаючи авторів даємо для ознайомлення рецензію, яка дає в цілому негативну оцінку такому дидактичному матеріалу (додаток 4).

РОЗДІЛ VII.**VII.1 ОРГАНІЗАЦІЯ ПОЗАКЛАСНОЇ РОБОТИ З ОСНОВ ІНФОРМАТИКИ**

Позакласна робота з будь-якого навчального предмету має *мету, завдання, особливості, напрямки (форми) та ступені (рівні)*. Мета – це досягнення ключового, стратегічного завдання: виявлення обдарованих учнів і їх максимальний розвиток у даній галузі. Решта *завдань* мають тактичний характер, вони впливають із ключового й тісно пов’язані між собою. Найважливіші серед них такі: популяризація навчального предмету і зацікавлення ним якомога більшої кількості учнів; виявлення рівня здібностей та підготовленості кожного учня; організація навчальної підготовки; створення атмосфери змагання.

Серед особливостей перш за все слід виділити поєднання гурткового характеру занять та індивідуальної форми навчальної роботи. *Особливостями* позакласної роботи з інформатики є тісні міжпредметні зв’язки, базування на математичній основі та прикладна спрямованість. Її результати – це абстрактні програмні проекти, реалізовані на комп’ютері. Але основну частину ку-

ресу інформатики складають інформаційно-комунікаційні технології, тому позакласна робота з інформатики логічно мало пов'язана з основним курсом.

Виходячи з мети, домовимось не відносити до позакласної роботи з інформатики масові заходи розважального характеру зразка вікторин, КВК тощо. Основними формами позакласної роботи вважаються підготовка та

схема VIII.1



участь у предметних олімпіадах і робота в секціях МАН. Але, якщо поміркувати, то це не напрямки, а *ступені* (див. схему VIII.1).

На місцях часто дотримуються абсурдного правила: відібрати зацікавлених учнів; здібніших спрямувати на розв'язування олімпіадних завдань; решті запропонувати написання рефератів для секцій МАН. Циклічність участі в роботі секції МАН переважно не більше одного навчального року. При таких умовах учень може бути лише озвучувачем (спікером) тез до реферату, адже серйозна дослідницька робота та виконання програмного проекту вимагає тривалого часу і кваліфікованих навичок у вузькій галузі.

Така участь у роботі МАН з одного боку є профанацією самої ідеї, з іншого – фальсифікацією результатів, нарешті своєрідним проявом корупції, адже пов'язана з присудженням призових місць та премій, встановленням рейтингів і пільгових прав для вступу на навчання. Вона не потрібна ні учневі, ні учителю, адже забирає час, приносячи замість практичної користі лише моральну шкоду. Як і будь-які фальсифікації, стверджувати доказово це дуже важко як з технічної, так і з правової сторін, але, враховуючи специфіку – навчально-виховну роботу з підростаючим поколінням, необхідно морально засуджувати. Як доказ, обмежимося констатацією кількох, але красномовних фактів. У конкурсах-захистах МАН беруть участь дослідницькі проекти, які доступні лише поодиноким, особливо обдарованим і підготовленим учням. Отже щорічна кількість учасників конкурсів-захистів може бути хоча б на

один порядок меншою за кількість учасників олімпіад з інформатики, а насправді спостерігається майже однакова кількість. Багато разів доводилось слухати захисти рефератів учнями, які явно не розуміли використаних ними термінів. Безперечно, що за кожною роботою МАН повинен стояти науковий керівник, яким може бути далеко не кожен шкільний учитель. У сільській місцевості роль наукових і педагогічних керівників майже виключно виконують учителі-предметники, причому, часто один керує кількома конкурсантами відразу.

Висновки очевидні, але не будемо поспішати їх робити. Справа у порушенні рівності умов таких конкурсів для елітних і пересічних навчальних закладів, для міської та сільської місцевості. Вищеописаними фактами та висновками не переслідувалась критична мета, вони потрібні у контексті роздумів про організацію позакласної роботи з інформатики лише як несприятливе явище.

Робота в секціях МАН може йти паралельно з поглибленим вивченням предмету, підготовкою до участі в олімпіадних змаганнях, але досягти успіху може лише при організації справжньої індивідуальної дослідницької роботи, що базується на глибокому знанні предмету і високому рівні інтелекту, які досягаються при розв'язуванні великої кількості задач з «родзинкою», тобто олімпіадних.

На наше переконання робота в секціях МАН повинна бути лише там, де є висококваліфіковані кадри наукових та педагогічних керівників. Захист рефератів слід проводити, починаючи з обласного етапу, бо в середньостатистичному районі важко знайти конкурентні між собою проекти.

Таким чином, правильно організована робота секцій Малої академії наук є вищим ступенем або рівнем позакласної роботи з інформатики, а сфера розв'язування олімпіадних задач з програмування у такому випадку стає першим, базовим рівнем і разом з цим найбільш серйозною та масовою формою позакласної роботи.

Правда тут також є важливе застереження. Воно полягає в тому, що програмування у загальноосвітніх школах вивчається дуже обмежено і лише на завершальному етапі навчання. Олімпіади ж проходять у листопаді-березні, коли навіть випускники ще не ознайомлені з курсом програмування. Таким чином олімпіади з інформатики (читай – з програмування), щоб не порушувати рівності умов, про що говорилось вище, доцільно проводити окремо для учнів спеціалізованих шкіл з поглибленим вивченням інформатики та спеціальних гуртків і факультативів, і окремо для учнів загальноосвітніх шкіл.

Але, всупереч цьому, вже на протязі двох десятиліть олімпіади з інформатики проводяться масово для всіх шкіл. Як показує практика, успішно виступають лише ті учні, які на час участі в олімпіаді вже 3-4 роки вивчали програмування поглиблено під керівництвом досвідчених викладачів.

Нижче розглянемо базову форму позакласної роботи з інформатики, тобто організацію підготовки та проведення олімпіад.

VII.2 ПРО СУЧАСНІ ОЛІМПІАДИ З ІНФОРМАТИКИ.

Питання про суть і завдання сучасних учнівських олімпіад з інформатики, коли, як та з чого починати підготовку до них, досить очевидне, але зовсім не риторичне. Відповідь має кілька складових. Пропедевтичну роботу з програмування бажано починати приблизно з сьомого класу, бо, з одного боку, в учнів необхідно встигнути сформулювати дуже струнку, чітку і об'ємну наукову систему знань, з іншого – необхідна математична культура мислення, яка з'являється не раніше цього віку.

Необхідно уточнити термін «учнівські олімпіади з інформатики», переглянувши форми і методи їх підготовки та проведення, тобто змінити акценти і узгодження із шкільним предметом «основи інформатики». Спробуємо сформулювати і узагальнити вимоги до олімпіад в традиційному розумінні (далі – просто *олімпіади з інформатики*) та згідно сучасних вимог накреслити шляхи олімпіадної роботи на нижчих (I-III) етапах. Розуміючи, що тема надто широка, обмежимося її фрагментарним оглядом і орієнтовними методичними порадами, не вдаючись до рекомендацій, які повинні спиратись на нормативну базу та педагогічні дослідження.

Як відомо, завдання не тільки для III-IV, а й для нижчих етапів, вимагають знання ряду тем, що виходять за межі шкільних програм з інформатики та математики. Як тут не пригадати слова з пісні: «...кандидат наук и тот над задачей плачет». Учням пропонуються завдання, виконання яких потребує не простої обізнаності, а свідомого розуміння і вільного використання відомостей з теорії множин, аналітичної геометрії, комбінаторики, теорії графів та інших математичних дисциплін, що вивчаються не в середніх навчальних закладах, а у вузах, причому не завжди на перших курсах і переважно на математичних та технічних спеціальностях. Стали необхідними також вміння користуватись рекурсивним методом, методами довгої арифметики, перебору та динамічного програмування, вміння оцінювати програми на ефективність та швидкодію. Наведемо одну характеру цитату 10-15-річної давності [22]: «...метод динамічного програмування для скінченного простору рішень справді досить простий, а його ідея зрозуміла навіть школярам з пересічною підготовкою. Але й годі вимагати від них самостійно додуматись за кілька годин до того, що було останнім словом науки лише 35 років назад». Як видно, вона стосується доступності для учнів методу динамічного програмування. Перш за все вона дуже суперечлива. Чи справді він досить простий, та ще й для школяра з пересічною підготовкою? А як же з «...годі вимагати від них самостійно додуматись за кілька годин»? І, врешті, навіщо учням, навіть під час олімпіад давати завдання, для розв'язування яких необхідно знати, розуміти і вміти застосовувати те, що було останнім словом науки лише кілька десятиріч тому? Очевидно, що олімпіади з інформатики з самого початку поміщені в логічний вакуум. Наше завдання – звузити його до тонкої і прозорої плівки, через яку пересічний учень може побачити горизонти програмування, а при бажанні чи потребі досягти їх. Тому позакласну роботу з інформатики слід вести якісно, як додатковий навчальний предмет, причому, не тільки для забезпечення успішної участі учнів у олімпіадах, а й для розвитку в них алгоритмічної культури, логічного мислення, комп'ютерної грамо-

тності, підготовки до навчання у вузах. Це проблема відповідності задач математичній підготовленості учасників олімпіади.

Окрема проблема – це *відсутність вікової диференціації*. Адже у програмуванні важко класифікувати задачі, як по віку, так і по складності необхідного теоретичного матеріалу. Через це на олімпіадах з інформатики здебільшого даються однакові задачі для 8-11 класів. Спробуємо знайти один із варіантів розв'язання цієї проблеми, навівши приклади завдань, окремих для 10 та 11 класів.

9-10 класи

- 1) *Системи числення (Notations10)*. Дано натуральне число X у четвірковій системі числення. Описати алгоритм його переведення у вісімкову систему числення.
- 2) *Просте число (SimplNumb10)*. Описати алгоритм перевірки, чи буде простим дане натуральне число.
- 3) *Сходишки (Ladder10)*. По східцях (21 сходинка) вгору піднімається десятикласник. Описати алгоритм підрахунку кількості способів дістатись до верхньої сходинки, якщо за один крок він може піднятись не більше, ніж на 3 сходинки.
- 4) *Обмін місцями (ExchangPlac10)*. Потрібно описати стратегію обміну місцями k значків « $\times$ » та « $\circ$ » на полі, що має вигляд смужки з $2k+1$ клітинок (див. мал.), якщо «хрестики» можуть рухатись тільки в напрямку вправо, а «нулики» можуть рухатись тільки в напрямку вліво, стаючи на сусідню клітинку, якщо вона вільна, або «перестрибнувши» через різнойменний значок, якщо за ним є вільна клітинка:

$\times$	$\times$	...	$\times$		$\circ$	$\circ$	...	$\circ$
----------	----------	-----	----------	--	---------	---------	-----	---------

11 клас

- 1) *Системи числення (Notations11)* Дано натуральне число X у n -ковій системі числення. Описати алгоритм його переведення у m -кову систему числення, якщо $m \neq n$ і $2 < n < 16$, $2 < m < 16$.
- 2) *Просте число (SimplNumb11)* Описати алгоритм підрахунку кількості простих чисел, які знаходяться між двома даними натуральними числами M та N .
- 3) *Сходишки (Ladder11)* По східцях ($k \leq 30$ - кількість сходинок) вгору піднімається одинадцятикласник. Описати алгоритм підрахунку кількості способів підняття на верхню сходинку, якщо за один крок він може піднятись не більше, ніж на n сходинок.
- 4) *Обмін місцями (ExchangPlac11)* Потрібно описати стратегію обміну місцями значків « $\times$ » та « $\circ$ » на квадратному полі з $(2k+1) \times (2k+1)$ клітинок (див. мал., $k=3$), якщо кожен «хрестик» може рухатись тільки вправо або тільки вниз, а кожен «нулик» може рухатись тільки вліво або тільки вгору, стаючи на сусідню клітинку, якщо вона вільна, або «перестрибнувши» через різнойменний значок, якщо за ним є вільна клітинка:

$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$
$\times$	$\times$	$\times$	$\times$	$\times$	$\times$	$\times$

×	×	×		○	○	○
○	○	○	○	○	○	○
○	○	○	○	○	○	○
○	○	○	○	○	○	○

Прокоментуємо наведені завдання. Перш за все в них замість вимоги написати програми згідно з даними технічними умовами, пропонується описати алгоритми, що важливо для теоретичного туру. Кожне завдання можна звести до прозорого, не дуже громіздкого алгоритму, опис якого наближений до звичайного програмного матеріалу з предмету, зводиться до формулювання ідеї, не потребує глибоких теоретичних знань. Нарешті, умови вибрано так, що завдання для 11-го класу є логічним ускладненням завдання для 10-го класу з таким же номером. На наш погляд, обидві сформульовані вище проблеми тут якщо не вирішені цілком, то, принаймні, зведені до мінімуму.

Тут не даремно описані зразки завдань, придатних для теоретичного туру олімпіади. Ми вважаємо, що на I-II етапах, коли регламент залежить від оргкомітету шкільного та районного масштабу, олімпіаду корисно проводити в два тури (I тур – теоретичний, II тур – практичний). При цьому на районному етапі олімпіади слід забезпечити проведення теоретичного туру за кілька днів до практичного, з використанням завдань, складених районним журі. Доцільність такої форми пов'язана з третьою проблемою, яка полягає у невідомості співвідношення між теоретичною та практичною частинами роботи над завданнями. З одного боку не можна ігнорувати роботу над описом математичної моделі, складанням, доведенням та оцінкою алгоритму, з іншого боку, тільки ступінь практичної реалізації у вигляді діючої програми дозволить зробити правильну оцінку.

Офіційні суперечки про те, в якій формі - теоретичній, практичній чи комбінованій слід проводити олімпіади з інформатики на III-IV етапах, давно вважаються завершеними на користь практичної. Тому перевірка письмових робіт і аналіз програм не обов'язкові. Єдиним визнаним продуктом учасника олімпіади є програма. Така позиція загальноприйнята. Переконливо сказано про це в [21]: «Акцент усе більше зміщується від «краси алгоритму» до «діючої програми»... Журі олімпіад уже не розглядає тексти програм учасників. Оцінювання діючої програми здійснюється як оцінювання діючого «чорного ящика»: на вхід подаються вхідні дані, а на виході має бути правильний результат. Хитромудрі обмеження, що містяться в умові, задають необхідний рівень ефективності розв'язання».

Якщо в умовах задач вимагається описати алгоритм, то, очевидно, постає ще одна проблема, не логічного, як три попередні, а технічного характеру: як описувати алгоритм. Цій проблемі присвяtimo нижче окремий пункт. Через відсутність однакових умов проведення на місцях, участь в олімпіадах з інформатики не може бути обов'язковою, що на практиці де-факто має місце. Отже дані про участь і результати олімпіад з інформатики не можуть бути обов'язковими компонентами в статистичній звітності та при підведенні підсумків роботи навчальних закладів.

Проведення районних олімпіад з інформатики за зразком обласного етапу не ефективно. Перш за все для організації районного туру, особливо у сільських районах, важко знайти єдиний населений пункт з достатньою кількістю комп'ютерів. Значна частина учнів, особливо 9-10 класів у листопаді-грудні, як уже згадувалось, ще не має достатніх навичок програмування, тому для значної (інколи переважної) кількості учасників практичний тур з використанням комп'ютерів не надто актуальний. Всі етапи Всеукраїнської олімпіади з інформатики мають спільну головну мету: відбір учасників наступного етапу. Але, очевидно, що будь-яка олімпіада покликана також популяризувати предмет, сприяти піднесенню загального рівня знань, умінь та навичок (див. схему VIII.1). У зв'язку з цим можна погодитись, що різні етапи олімпіади з інформатики повинні мати власні локальні завдання. Наприклад, перший (шкільний) етап у першу чергу покликаний виявити тих, хто цікавиться предметом більше за інших та володіє елементами програмування. Другий (районний) етап може відібрати серед призерів I-го етапу тих, хто має здібності до точних наук, схильність абстрактно, логічно, структурно та алгоритмічно мислити, має певні навички програмування. Третій (обласний) етап дає змогу відібрати учнів, які мають значний досвід програмування та перспективи удосконалення. IV-й етап виявляє обдарованих у галузі програмування. Умовно можна об'єднати у дві групи I-II та III-IV етапи. Очевидно, враховуючи це, послідовно від етапу до етапу слід змінювати методiku підготовки завдань та умови проведення змагань. Так на IV-у етапі склалась традиція проводити два практичні тури. На наш погляд, як уже було сказано вище, на I-II етапах резонно також проводити два тури, але перший – теоретичний, а другий – практичний.

Олімпіади – це марафон для особливо сильних. Тому після теоретичного туру до наступних змагань слід допускати лише порівняно невелику кількість здатних до такої специфічної сфери діяльності, як програмування. Проведенню районних етапів олімпіади з інформатики є кілька перепон. Серед них першою можна вважати включення районних олімпіад з інформатики до етапів Всеукраїнської олімпіади, а це не дозволяє змінювати умови проведення. Проте ця перепона легко усувається – можна вважати теоретичний тур попереднім і неофіційним. Залишається лише дозволити в другому турі не брати участь навчальним закладам, які не можуть виставити достатньо підготовлених учасників.

Такі зміни дозволяють проводити практичний тур районної олімпіади за всіма правилами, як зауважено в наведеній вище цитаті [21], можна буде змістити акцент від «краси алгоритму» до «діючої програми», а журі одержить можливість виключно оцінювати дієвість програми, застосувавши в тестах обмеження, що містяться в умові і задають необхідний рівень ефективності розв'язання.

Існує ще одна, можливо, не дуже суттєва перепона впровадженню теоретичного туру. Справа в тому, що графік олімпіад та конкурсів-захистів у секціях МАН досить тісний, тому важко організувати два тури олімпіади з інформатики.

Результати районних олімпіад слід розглядати на семінарі вчителів інформатики, підготувавши методичні рекомендації. Вони можуть бути посібником для вчителів і учнів для наступної позакласної роботи з предмету.

VII.3 ЯКОЮ ПОВИННА БУТИ ОЛІМПІАДНА ЗАДАЧА? ПИСЬМОВИЙ ОПИС АЛГОРИТМА ТА ЕТАПИ РОЗВ'ЯЗУВАННЯ

Якщо є діюча при всіх умовах і обмеженнях задачі програма, то ніякі правила опису алгоритмів та оформлення письмової частини роботи не потрібні. Але коли немає діючої програми, доведеться оцінювати написане учнем, інакше перевірка буде формальною.

Після аналізу письмової частини роботи учасника олімпіади з інформатики можна об'єктивно оцінити роботу, як за наявності програми, діючої коректно чи ні, так і без програми взагалі.

Що повинно обов'язково входити до письмової частини олімпіадної роботи? Назвемо та опишемо орієнтовні етапи:

Дослідження умови задачі та оголошення й опис величин;

Опис математичної моделі;

Загальний опис і доведення алгоритму;

Уточнення алгоритму та оцінка його складності;

Щоб розібратись, як доцільно підготувати письмову частину олімпіадної роботи з інформатики, можна рекомендувати ретельно опрацювати дуже цінні в цьому відношенні посібники [23] та [24].

Тут для ілюстрації наведемо інший приклад.

Задача Num. (в більш ускладненому варіанті пропонувалась у 2000-2001 н.р. на обласній олімпіаді в Київській області) Дано натуральне число N , записане n одиницями. Визначити число k - найменшу кількість чисел, в записі яких є лише цифри «3» і які в сумі дорівнюють числу N .

Технічні умови:

вхідний файл *Num.dat* містить натуральне число n ($1 < n < 255$).

вихідний файл *Num.res* повинен містити число k .

a) Дослідження умови задачі та оголошення й опис величин.

1) Згідно умови задачі аргументом повинне бути число n цілого типу ($1 < n < 255$), результатом - число k (найменш можлива кількість доданків, також цілого типу).

2) Згідно ознаки подільності натурального числа на 3 робимо висновок: дане число N представляється у вигляді суми, кожен доданок якої містить лише цифри «3» тоді і тільки тоді, коли кількість цифр числа N ділиться без остачі на число 3. Отже, якщо n ділиться без остачі на 3, то представлення його згідно умови задачі можливе, інакше – ні.

3) Повний перебір всіх варіантів запису суми чисел, що мають у записі лише цифри «3» через обмеження $1 < n < 255$ неефективний.

b) Опис математичної моделі.

Для відшукування алгоритму спочатку покладемо $n = 5$ (тобто $N = 11111$) і помітимо, що воно представляється згідно умови задачі у вигляді:

$$N = \underbrace{(3333+3333+3333)}_{9999} + \underbrace{(333+333+333)}_{999} + \underbrace{(33+33+33)}_{99} + \underbrace{(3+3+3)}_9 + \underbrace{3+3}_{\text{остача}} \quad (1)$$

основна частина суми

Як видно з (1), число N має 14 доданків, тобто $k = 14$. *Основною частиною суми* 4 (кількість груп доданків, що мають у записі цифру «3»).

Наведена формула може бути узагальнена, і матиме вигляд:

$$N = \underbrace{(33\dots3+33\dots3+33\dots3)}_{\text{число } n \text{ цифр "3"}} + \dots + \underbrace{(33333+33333+33333)}_{99999} + \underbrace{(3333+3333+3333)}_{9999} + \underbrace{(333+333+333)}_{999} + \underbrace{(33+33+33)}_{99} + \underbrace{(3+3+3)}_9 + \text{остача} \quad (2)$$

Остача – це число, менше за 9, воно може ділитись на 3 без остачі, або мати остачу 1 чи 2. В першому випадку це число додасть до числа k одиницю або двійку, у другому приведе до відповіді: $k = 0$. Очевидно, що остання формула дозволяє отримати найменш можливе значення числа k , тобто цілком відповідає умові задачі і може бути її математичною моделлю.

с) *Загальний опис і доведення алгоритму.*

Алгоритм розв'язування задачі у загальному вигляді буде такий:

алг NUM

арг ціл n

рез ціл k

поч ціл *основна_частина_суми, остача*

основна_частина_суми }
остача } згідно формул (1) та

ввод n

якщо $\text{mod}(n, 3) \neq 0$

то $k:=0$

інакше *основна_частина_суми* := $3(n - 1)$

визначити(остача) | допоміжний алгоритм

$k := \text{основна_частина_суми} + \text{div}(\text{остача}, 3)$

все

вивод k

кін

Прокоментуємо форму запису алгоритму. Вона може бути довільною, в тому числі словесною чи у вигляді блок-схеми. Головне, щоб вибраний спосіб опису алгоритму був використаний доцільно й грамотно. Нам здається, що краще дотримуватись правил запису на алгоритмічній мові. При цьому слід привчати учнів дотримуватись позначень, введених у а) та б) що в коментарях (на наш погляд, обов'язкових) допоможе уникнути багатослів'я. Коментарі слід використовувати, як для опису величин, так і в тілі алгоритму.

При оголошенні величин бажано використовувати словесні імена, вживаючи знак «_» для з'єднання кількох слів в одне ім'я, наприклад: *основна_частина_суми*.

Під основним алгоритмом слід повністю за такими ж правилами навести допоміжні алгоритми, записані у вигляді *процедур* чи *функцій*.

Доведення існування та єдиності алгоритму – це *теорема*, тому воно записується згідно традиційних правил доведення теорем. Але спочатку слід чітко сформулювати умову й висновок, які важливо не сплутати з аргументами й результатами задачі. Для нашої задачі теорему існування її розв’язання можна сформулювати так:

Теорема NUM. Якщо дано натуральне число N , записане n одиницями ($1 < n < 255$), то існує алгоритм, який визначає число k – найменшу кількість чисел, в записі яких є лише цифри «3» і які в сумі дорівнюють числу N . (умова й висновок виділені курсивом)

Доведення впливає з 2), (1) та (2). Дійсно, ця формула представляє число N , як суму найменшої кількості доданків, записаних дев’ятками, а кожен доданок, у свою чергу, виділяє найменш можливу кількість чисел, записаних трійками. *Остача* – це число, менше 9 і з нього легко визначити кількість доданків, рівних 3 із допомогою функції $\text{div}(\text{остача}, 3)$. Нарешті, процес віднімання від числа N доданків, записаних лише з допомогою цифр «3» скінченний. Ці міркування повністю доводять теорему NUM.

При підготовці учнів до олімпіад з інформатики важливо уточнити також і поняття «олімпіадна задача». Чим же особливі олімпіадні задачі з інформатики (програмування)? Перш за все вони значно «важчі» від тих, які передбачено розглядати згідно програми вивчення курсу інформатики. Усім відомо висловлення академіка А.П. Єршова «Інформатика йде від математики». Тому задачі слід класифікувати перш за все за типами математичних моделей, тобто відповідністю певним розділам математики. Під основними типами олімпіадних задач за математичними моделями тут розуміються традиційні задачі на повний і удосконалений перебір, рекурсії, методи оптимізації та динамічного програмування, координатний метод, графи, відшукування виграшних стратегій і т.д. (див. [22]). Ефективне вивчення евристики в інформатиці можна провести з допомогою посібника [24].

По рівню складності задачі можна поділити на три типи: прості («втішальні»), технічні та складні («непідійомні»). Очевидно, що рівень складності – поняття відносне для різних рівнів олімпіади. Особливо слід виділити так звані задачі-пастки. Вони водночас можуть бути й простими й підвищеної складності. Справа в тому, що при знайденні вдалої математичної моделі така задача стає занадто легкою з точки зору програмування. Як приклад можна навести задачу ([25], IV етап XIII Всеукраїнської олімпіади з інформатики):

Дано послідовність, що складається з $2N$ натуральних чисел, які можна розбити на пари таким чином, що сума чисел у всіх парах буде однаковою. Наприклад, числа послідовності 99, 23, 77, 1 можна поділити на пари $1+99=77+23$. Написати програму, яка для даної послідовності встановлює, чи можна її розбити на пари таким чином, щоб добуток чисел у всіх парах був також однаковим.

Після застосування в математичній моделі відомої всім учням теореми Вієта програма розв'язання цієї задачі записується буквально в кілька рядків.

На олімпіаді одночасно слід пропонувати задачі різного рівня складності, але ні в якому разі не можна давати задач, доведених за рахунок обмежень в умовах «до абсурду», що в попередні роки часто траплялось, особливо на IV етапі.

Можна порадити таку послідовність підготовки:

- Повторення й узагальнення традиційного матеріалу про алгоритми та програми, способи їх запису, основні структури алгоритмів, етапи розв'язування задач на комп'ютері;
- Вивчення основ алгоритмічної мови та мови програмування з розглядом і узагальненням понять числа, величини, процедури, функції, перетворення типів в інформатиці;
- Розв'язування так званих «технічних» задач, яке вимагає мінімум зусиль для побудови математичної моделі задачі, але дозволяє відпрацювати «до автоматизму» навички оперування циклами, розгалуженнями, масивами, процедурами, функціями, кодуванням, файлами вводу/виводу, ресурсами комп'ютера (пам'яттю й швидкодією), директивами компілятора, динамічною пам'яттю і т.д.;
- Розв'язування основних типів олімпіадних задач (за математичними моделями);
- Розв'язування задач на евристику.

У якості «технічних» можна пропонувати задачі на пошук, сортування «довгу арифметику».

При підготовці до олімпіади замість терміну «кінцевий етап» краще вживати «зрілий етап». На ньому пропонуються комбіновані в розумінні типів задачі, а учні *обов'язково* повинні вміти працювати у автономному режимі.

VII.4 ЗАМІСТЬ ПІСЛЯМОВИ.

Вище в оглядовому плані описано зміст роботи обласної школи передового педагогічного досвіду при Київському обласному інституті післядипломної освіти педагогічних кадрів у 2007-2010 роках. Щоб відтворити роботу ОШПД повністю, необхідно навести всі річні плани роботи та плани проведення кожного із 12 занять школи (у додатку наведено лише розділ «Практична діяльність»), зміст понад 10 прочитаних лекцій, приблизно 25 проведених практичних та семінарських занять, які не втратять актуальності протягом наступних 3-5 років, тому разом можуть утворити авторські курси підвищення кваліфікації для учителів основ інформатики. До названих матеріалів слід додати велику кількість творчих домашніх завдань найактивніших слухачів, які подавались у електронному вигляді та частково у формі брошур формату А5 обсягом 10-20 сторінок кожна. Але специфіка методичної роботи учителя інформатики така, що більшість дидактичних матеріалів мають демонстраційний електронний формат (наприклад, навчальні презентації), який втрачає основний зміст у вигляді, придатному для друку, тому лише кращі

зразки цих матеріалів буде запропоновано для публікації у щорічному методичному збірнику КОІПОПК.

Робота ОШППД мала конкретний і актуальний характер, що впливав із повсякденних потреб учителів інформатики. Для вивчення піднімалися проблеми, з одного боку глобального характеру, з іншого цілком доступні для вирішення на місцях силами самих учителів.

Проте, складне поєднання ініційованих проблем має динамічний характер, тому їх розв'язання у повному обсязі можливе лише в ідеалі. Виходячи з цього, керівник ОШППД, допомагаючи слухачам у практичному вирішенні названих вище шести актуальних проблем, намагався також озброїти їх дієвими методичними інструментами, з допомогою яких слухачі у майбутній практиці зможуть самостійно виявляти актуальні питання та шукати способи їх розв'язування. Іншими словами, протягом трьох років кожен активний слухач не тільки і не стільки отримував фахову та методичну допомогу, він також, що головне, ставав у доступній мірі методистом-практиком.

Предмет «основи інформатики» має специфіку, яка полягає у дуже обмеженій кількості навчального часу (всього 70 уроків для універсального профілю) та методичній «віддаленості» від традиційного кола навчальних предметів, що створює не тільки серед учителів, адміністративних та методичних працівників, а й багатьох учнів та їх батьків ілюзорне ототожнення шкільної інформатики із побутовими навичками роботи з комп'ютером, де сьогодні більшість вважають себе спеціалістами. Тому часто вважається, що наявність у школі комп'ютерного класу може автоматично вирішити всі проблеми інноваційних та інформаційно-комунікаційних технологій навчання, використання інтернету, створення шкільних сайтів, забезпечення дистанційного навчання та інтерактивного зв'язку з батьками тощо. Всі ці проблеми неправомерно «зав'язуються» на учителеві, який часто викладає інформатику по сумісництву. В роботі ОШППД піднято лише проблеми, що стосуються учителів інформатики, для вирішення щойно названих проблем, які слід адресувати перш за все адміністраціям шкіл та планово-фінансовим органам, необхідні значні фінансування у вигляді оплат послуг спеціальних установ для обслуговування комп'ютерної техніки, послуг інтернету, закупівлі прикладного програмного забезпечення та сервісних програм, при наявності двох комп'ютерних класів у навчальному закладі обов'язкова оплата обслуговуючого персоналу, в тому числі лаборанта та інженера. Замість цього практикується тенденція скорочення ставок лаборантів в кабінетах ІКТ, відміна доплат за шкідливість викладачам інформатики. В таких умовах учитель інформатики, який повинен бути головним спеціалістом навчального закладу з ІКТ, не в змозі ні підтримувати свою фахову кваліфікацію, ні вирішувати актуальні проблеми шкільної інформатики.

ДОДАТКИ

ДОДАТОК I

ПЛАНУВАННЯ РОБОТИ ОШПІД ПРОТЯГОМ 2007-2010 Р.Р.

(розділ "Практична діяльність")

2007-2008 н.р.

		види занять			
№	тема занять	зміст занять	лекція	семінар	практика
I	Сучасні проблеми у викладанні основ інформатики та шляхи їх розв'язання .				
1	Якою бути шкільній інформатиці сьогодні і завтра? ⁱ ;		1,5 год.		
2	Презентація досвіду викладання основ інформатики в Щасливському НВК		1,5 год.		
3	Роль теми «Алгоритмізація і програмування» в курсі основ інформатики		1,5 год.		
4	Інформатика іде від математики. Домашнє завдання для слухачів: <i>Скласти план використання додатка Windows «Калькулятор» при вивченні теми «Кодування та вимірювання інформації»</i>				1,5 год.
II	Місце інформаційно-комп'ютерних технологій у викладанні основ інформатики				
1	Комплексні електронні дидактичні засоби на уроках основ інформатики ⁱⁱ ;		1,5 год.		
2	Обговорення та рецензування комплексу таблиць та дидактичної довідкової системи «Турбо Паскаль у таблицях»			1,5 год.	
3	Тренінг «Моделювання електронного сценарію уроку». Домашнє завдання для слухачів: <i>Скласти електронний сценарій уроку по темі «ОС Windows»</i>				1,5 год.
4	Тренінг «Розробка електронних дидактичних засобів з допомогою Microsoft Power Point». Домашнє завдання для слухачів: <i>Скласти електронний конспект уроку по самостійно вибраній темі з основ інформатики для 10 класу</i>				1,5 год.
III	Організація позакласної роботи з основ інформатики				
1	Якою бути олімпадній задачі з інформатики? Вимоги до умови, тестів, опису математичної моделі та програми		1,5 год.		

2	Метод динамічного програмування у школі	1,5 год.
3	Розв'язування задачі методом динамічного програмування. Домашнє завдання для слухачів: <i>Підготувати для колективного збірника задач на метод динамічного програмування одну задачу з теоремами та авторськими розв'язком</i>	2 год.
4	Зустріч із редколегією газети «Інформатика»	2 год.
IV Застосування методу проектів у процесі викладання основ інформатики		
1	Місце методу проектів в інтерактивних технологіях навчання.	1,5 гол.
2	Роль методу проектів у підготовці та проведенні уроків-семінарів з основ інформатики	1,5 год.
3	Організація підготовки до державної підсумкової атестації учнів з основ інформатики як проект.	1,5 год.
4	Творчі звіти слухачів ОШППД (по домашніх завданнях, одержаних на попередніх заняттях)	3 год.
<i>Систематично проводити консультації:</i>		
- Консультації щодо проведення уроків та позакласних занять з основ інформатики;		
- Консультації щодо підготовки до занять ОШППД		

2008-2009 н.р.

№	тема занять	зміст занять	види занять		
			лекція	семінар	практика
I Викладання курсу “Алгоритмізація та програмування“ у формі експрес-курсу. .					
1	Презентація плану роботи ОШППД на 2008-2009 н.р.		0,5 год.		
2	Лекція: «Методичні засади експрес-курсу «Алгоритмізація та програмування» (ЕКАП)».		1,5 год.		
3	Вступні теми ЕКАП-у.			1 год.	
4	Створення зразка уроку ЕКАП на вибрану тему у формі електронного конспекту уроку.				1,5 год.
5	Інструктаж по виконанню творчих домашніх завдань. Завдання: <i>Підготувати перелік понять, питань, прикладів та завдань до окремої вибраної теми ЕКАПу.</i>			1,5 год	.
II Методична і дидактична робота учителя інформатики.					
1	Лекція: «Організація методичної і дидактичної роботи учителя інформатики. Форми, обсяги та цикли».		1,5 год.		
2	Створення проекту «Олімпіадна задача з інформатики».				1,5 год.
3	Захист творчих домашніх завдань слухачів на тему «Поняття, питання, приклади та завдання до уроку ЕКАП»			1,5 год.	
4	Інструктаж по виконанню творчих домашніх завдань. Завдання: <i>Підготувати у формі брошури формату А5, обсягом 12-16 сторінок, розміром шрифту 10 проект «Задачі до I етапу Всеукраїнської олімпіади з інформатики».</i>			1,5 год.	
III Математична складова курсу основ інформатики					
1	Лекція по міжпредметних зв'язках інформатики з математикою: «Що дає інформатиці математика і математиці інформатика?»		1,5 год.		
2	Бінарний урок інформатики і математики.				1,5 год.
3	Захист творчих домашніх завдань слухачів на тему «Задачі до I			1,5	

	етапу Всеукраїнської олімпіади з інформатики». Створення колективного збірника олімпіадних задач.	год.	
4	Інструктаж по виконанню творчих домашніх завдань. Завдання: <i>Підготувати у формі брошури формату А5, обсягом 12-16 сторінок, розміром шрифту 14 п. серію з 2-3 бінарних ЕКУ по темі «Застосування електронних таблиць»</i>	1,5 год.	
IV	Творчий звіт обласної школи передового педагогічного досвіду за 2008 - 2009 н.р.		
1	Презентація творчого доробку ОШППД за 2008-2009 н.р.	1,5 год.	
2	Рецензування матеріалів слухачів ОШППД (по 2 брошури від кожного слухача згідно II-го та III-го занять).		1,5 год.
3	Семінар «Розгляд матеріалів творчого доробку учасників ОШППД в 2008-2009 н.р.»	3 год.	
4	Підсумки роботи ОШППД за 2008-2009 н.р.	1 год.	
	<i>Систематично проводити консультації:</i>		
-	Консультації щодо проведення уроків та позакласних занять з основ інформатики;		
	- Консультації щодо підготовки до занять ОШППД		

2009-2010 н.р.

№	тема занять	зміст занять	види занять		
			лекція	семінар	практика
I	Особливості викладання основ інформатики в 9-у класі за новою програмою та відповідними підручниками.				
1	Презентація плану роботи ОШППД на 2009-2010 н.р.		0,5 год.		
2	Лекція: «Особливості нової програми з основ інформатики та рекомендованих підручників для 9 класу».		1,5 год.		
3	Вимоги до методичних матеріалів слухачів ОШППД.			1 год.	
4	Порівняльна характеристика нових підручників з основ інформатики для 9 класу.				1,5 год.
5	Інструктаж по виконанню творчих домашніх завдань. Групове завдання: Підготувати експрес-курс для вивчення однієї з тем та методичні рекомендації щодо його використання відповідно нової програми з основ інформатики для 9 класу згідно вибраного підручника.			0,5 год	.
II	Методичні особливості вивчення ком-п'ютерних мереж в 9 класі				
1	Лекція: «Особливості дистанційної підготовки учнів 9-х класів до олімпіад з інформатики».		1 год.		
2	«Поєднання вивчення локальної мережі та інтернету в 9 класі».				1,5 год.
3	Обговорення та захист творчих домашніх завдань слухачів на тему «Локальні комп'ютерні мережі» та «Основи інтернету»			1,5 год.	
4	Інструктаж по виконанню творчих домашніх завдань. Завдання: Підготувати до друку експрес-курс та методичні рекомендації щодо його використання згідно завдання I заняття.			1 год.	
III	Методичні особливості вивчення векторної графіки в 9 класі.				
1	Лекція «Векторна графіка пакету Microsoft Office та її використання в курсі інформатики для 9 класу»		1,5 год.		
2	Презентація матеріалів слухачів, підготовлених до друку.			2 год.	
3	Інструктаж по виконанню творчих домашніх завдань. Завдання: Підготувати власні методичні доробки, створені протя-			1 год.	

гом навчання в ОШППД для заключного творчого звіту у формі презентацій та брошур.

IV Творчий звіт обласної школи передового педагогічного досвіду за 2007 - 2010 н.р.

- | | | |
|---|--|------|
| 1 | Презентація творчого доробку ОШППД за 2007-2010 н.р. | 1,5 |
| | | год. |
| 2 | Підсумки роботи ОШППД за 2009-2010 н.р. | 1,5 |
| | | год. |

Систематично проводити консультації:

- Консультації щодо проведення уроків та позакласних занять з основ інформатики;
- Консультації щодо підготовки до занять ОШППД

ДОДАТОК 2

ПРАКТИКУМ З ПРОГРАМУВАННЯ

Практична робота 1. ОБЧИСЛЕННЯ ЗНАЧЕННЯ МНОГОЧЛЕННА ЗА СХЕМОЮ ГОРНЕРА.

Спочатку ознайомимось із програмою швидкого обчислення значень многочлена за схемою Горнера, що часто буває дуже корисним. Нехай дано многочлен:

$$P(x) = a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-2}x^2 + a_{n-1}x + a_n,$$

(1)

де коефіцієнти позначені так, що сума індексу та відповідного степеня змінної рівна n . Для прикладу візьмемо: $P(x) = 11x^5 + 7x^4 - 2x^2 + x - 8$.

(2)

Складемо таблицю коефіцієнтів для (2), включаючи нулі, у порядку зростання індексів:

i	0	1	2	3	4	5
a_i	11	7	0	-2	1	8

Тепер перетворимо многочлен таким чином:

$$\begin{aligned} P(x) &= 11x^5 + 7x^4 + 0x^3 - 2x^2 + x - 8 = (11x^4 + 7x^3 + 0x^2 - 2x^1 + 1)x - 8 = \\ &= ((11x^3 + 7x^2 + 0x - 2)x + 1)x - 8 = (((11x^2 + 7x + 0)x - 2)x + 1)x - 8 = \\ &= (((11x + 7)x + 0)x - 2)x + 1)x - 8. \end{aligned}$$

(3)

Для остаточного варіанту (3) при даному значенні x алгоритм обчислення значення $P(x)$ полягає у виконанні циклу для значень $0 \leq i \leq n$, тіло якого містить множення попереднього значення (спочатку рівного a_0) на x та додавання a_i , що реалізовано функцією $S_G \{2\}$.

Крім того, представлення многочлена за схемою Горнера дозволяє уникнути операції піднесення до степеня і зменшити кількість арифметичних операцій при обчисленні його значення. Наприклад, (2) містить 16 операцій, а (3) тільки 10, різниця – 6 операцій. При великому n і незначній кількості коефіцієнтів, рівних нулю, ця різниця може бути дуже великою. Програма має вигляд:

```

program Gorners_Circuit;
  type mas=array[0..100] of real;           {1}
  var i,n:byte; x,y:real; A:mas;
  function S_G(n:integer;x:real;B:mas):real; {2}
    var i:byte;y:real;
  begin
    y:=B[0];                               {3}
    for i:=1 to n do y:=y*x+B[i];          {3}
    S_G:=y
  end;
begin
  repeat
    WriteLn('n?');ReadLn(n);
  until n<=100;
  WriteLn('x?');ReadLn(x);
  for i:=0 to n do begin
    WriteLn('A['i,']?');
    ReadLn(A[i]) end;
  WriteLn(S_G(n,x,A):12:2);
end.

```

! Слід звернути увагу на параметр *B* у заголовку функції *S_G*. З опису видно, що це масив (див {1}, {2}). Наголошуємо, що так описуються формальні параметри-масиви, які передаються у підпрограми. Раніше ми намагались уникати формальних параметрів у підпрограмах, але ПДС вимагають їх використовувати, бо це покращує читабельність програм і контроль за величинами під час їх трасування, крім того, допомагає уникати помилок виконання при одночасному використанні тих же глобальних величин програмою та її підпрограмами.

Корисно записати та зрозуміти загальну формулу (4) розкладу многочленна (1):

$$\begin{aligned}
 P(x) &= a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-2}x^2 + a_{n-1}x + a_n = \\
 &= ((\dots(a_0x^{n-1} + a_1)x + a_2)x + \dots + a_{n-1})x + a_n
 \end{aligned}
 \tag{4}$$

? *Питання до практичної роботи:*

1.1 Від чого залежить різниця у кількості операцій при записі многочлена у формах (1) та (4)?

1.2 Яка максимальна різниця в кількості операцій при обчисленні значення многочлена, представленого у формах (1) та (4)?

✎ Завдання 1.1 Скласти таблицю коефіцієнтів для розкладу за схемою Горнера многочленна: $P(x) = -3x^9 + 17x^7 - 12x^6 + 5x^4 + 9x^3 - 11x^2 + 16x - 18$.

<i>i</i>	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
<i>a_i</i>	1	2	3	0	4	0	5	6	7	0	0	8	0	9	10	11	12

Завдання 1.2 Написати многочлен, що відповідає таблиці коефіцієнтів:

Завдання 1.3 Змінити програму *Gorners_Circuit*, підключивши текстові файли для вводу/виводу.

Завдання 1.4 Вставити у програму *Gorners_Circuit* команди для підрахунку k – кількості арифметичних операцій при обчисленні значення многочленна і встановити залежність k від n та кількості коефіцієнтів, рівних 0.

Завдання 1.4 Записати програму *Gorners_Circuit* у вигляді функції *Gorners_Circuit_Func*.

Практична робота 2. ТАБУЛЮВАННЯ ФУНКЦІЇ.

Табулювання функції – це визначення її значень для значень аргументу на проміжку $[a, b]$ з кроком табуляції h , що можна використати для її дослідження та побудови графіка. Для цього необхідно ввести значення a , b та n – кількість значень x на $[a, b]$. Обчислення значення функції зручно записати у вигляді підпрограми-функції $F(x)$. Нижче наведено програму **Tab_Func** для функції $y=x/(1+x)$. Щоб скоротити пояснення, у програмі застосовано поширені коментарі. Але окремі фрагменти програми потребують детальнішого пояснення, вони виділені з допомогою додаткових коментарів у вигляді числа в фігурних дужках. Самі коментарі наведено нижче.



Приклад 2.1

```
program Tab_Func;                                {табулювання функції  $y = x/(1+x)$ }
  var a,b,x,h,y:real;
      n:integer;
  function F(x:real):real;
  begin
    if 1+x<>0                                     {1} {перевірка належності  $x$  ОДЗ}
    then F:=x/(1+x)                               {2} {обчислення значень функції}
    else F:= -1000000                             {3} {для  $x \notin$  ОДЗ,  $y = -1000000$ }
    end;
  begin
    WriteLn('a?');ReadLn(a);                      {ліва межа відрізка табулювання}
    WriteLn('b?');ReadLn(b);                      {права межа відрізка табулювання}
    WriteLn('n?');ReadLn(n);                      {кількість кроків табулювання}
    h:=(b-a)/n;                                    {4} {крок табуляції}
    x:=a;                                           {початкове значення  $x$ }
    while x<=b do begin
      WriteLn('y(',x:4:2,')=',F(x):8:4);          {вивод значень функції}
      x:=x+h                                         {збільшення значення  $x$  на крок}
    end
  end.
```

При заданому значенні n може трапитись, що x не належить ОДЗ аргументу. Тоді відповідне значення функції не можна обчислити. Для уникнення цього у підпрограмі-функції використана команда розгалуження {1}. Але підпрограма-функція обов'язково повинна мати єдиний результат, тому у випадку виходу значення x з ОДЗ підпрограма-функція в якості результату видасть завідомо не правильне значення, наприклад -10000000 {3}.

x	-1	-0,5	0	0,5	1	1,5	2	таблиця 2.1
$y(x)$	-1000000	-1,0000	0,0000	0,3333	0,5000	0,6000	0,6667	

Результати роботи програми для $a = -1$, $b = 2$, $n = 6$:

Змінною частиною програми є лише рядки {1} та {2}.

Скласти програму розв'язування рівняння $y = kx + b$ для

 **Приклад 2.2** цілих значень аргументу з проміжку $[-5; 5]$.

У наступній програмі, яка по суті аналогічна попередній (приклад 2.1), немає потреби визначати крок табуляції.

```

program Equation; {розв'язування рівняння  $y=kx+b$ }
var x, y:integer;
    k, b:real;
function F(x:integer):real;
begin
    F:=k*x+b {обчислення значень y}
end;
begin
    WriteLn('k?');ReadLn(k); {ввод коефіцієнта}
    WriteLn('b?');ReadLn(b); {та вільного члена рівняння}
    x:= -5; {початкове значення x}
    while x<=5 do begin
        WriteLn('(',x,', ',F(x),')'); {вивод розв'язків}
        x:=x+1 end {наступне значення x}
    end.

```

Результати роботи програми для $k = 2$ та $b = 3$:

x	-5	-4	-3	-2	-1	0	1	2	3	4	5	таблиця 2.2
y	-7	-5	-3	-1	1	3	5	7	9	11	13	

? Питання до практичної роботи:

2.1 Чи можна у програмі *Tab_Func* замість аргумента n вказати аргумент h (крок табуляції)?

2.2 Чому у програмі *Tab_Func* у випадку, коли $x \notin \text{ОДЗ}$, не можна вивести текстове повідомлення, наприклад “значення у обчислити не можна”?

2.3 Чи можна по таблиці 2.2 визначити кількість розв'язків рівняння $kx+b=0$?

2.4 Чи можна застосувати програму *Equation* застосувати для розв'язування нелінійних рівнянь з двома невідомими? Якщо можна, то у яких випадках?

 **Завдання 2.1** Написати програму, яка з допомогою функції *Gorners_Circuit_Func* виконує табулювання функції:

$$P(x) = a_0 x^n + a_1 x^{n-1} + a_2 x^{n-2} + \dots + a_{n-2} x^2 + a_{n-1} x + a_n.$$

Завдання 2.2* Написати програму *P_Equation*, яка з допомогою програми *Equation* та функції *Gorners_Circuit_Func* визначає кількість розв'язків рівняння:

$$a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-2}x^2 + a_{n-1}x + a_n = 0.$$

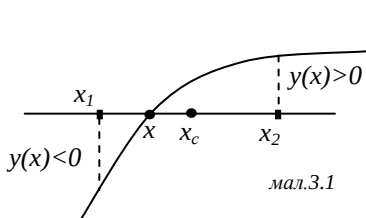
Завдання 2.3* Написати програму, яка визначає усі відрізки $[k; k+1]$, що містять корені рівняння $a_0x^n + a_1x^{n-1} + a_2x^{n-2} + \dots + a_{n-2}x^2 + a_{n-1}x + a_n = 0$.

Практична робота 3. УТОЧНЕННЯ РОЗВ'ЯЗКУ РІВНЯННЯ.

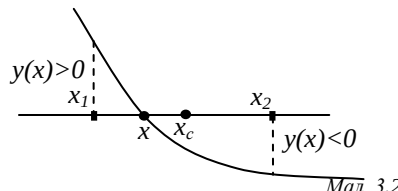
Після розгляду практичних робіт 1–2 та виконання завдань 2.2*–2.3* логічно поставити питання про *уточнення розв'язку* рівняння, адже існує багато типів рівнянь, у тому числі й рівнянь $p(x)=0$, де $p(x)$ – многочлен n -го степеня, для яких не існує аналітичних методів розв'язування.

① Уточненням розв'язку рівняння з точністю до $\varepsilon > 0$ (епсілон) називають визначення його наближеного значення x_0 , яке відрізняється від точного значення розв'язку x не більше, ніж на ε , тобто $x - x_0 \leq \varepsilon$.

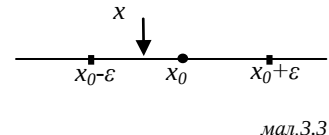
Нехай відомо, що на відрізку $[x_1, x_2]$ існує рівно один розв'язок x рівняння $y(x)=0$, де $y(x)$ – *неперервна* крива лінія. Тоді вигляд графіка функції $y(x)$ у ε -околі значення x (відрізок $[x-\varepsilon, x+\varepsilon]$, див. мал.3) матиме вигляд, показаний на мал.1, або на мал.2. *Характеристичною властивістю графіка y*



мал.3.1



Мал. 3.2



мал.3.3

околі точного значення розв'язку є те, що $y(x_1)$ та $y(x_2)$ мають різні знаки, тобто $y(x_1) \cdot y(x_2) < 0$.

Для визначення наближеного розв'язку рівняння можна застосувати відомий з уроку X метод половинного ділення, визначаючи значення $x_c = (x_2 - x_1)/2$ і перевіряючи, чи істинна нерівність $x_2 - x_1 \leq \varepsilon$. У випадку її істинності можна вважати x_c наближеним значенням розв'язку рівняння. У іншому випадку слід зменшити відрізок $[x_1, x_2]$, дотримуючись *характеристичної властивості графіка y у околі точного значення розв'язку*. Отже, програма уточнення розв'язку рівняння повинна містити щойно описані дії.

```

program SpecDec;                                {f(x)=3x3+4}
var a,b,c,x,e:real;
function F(e:real;var a,b:real):real; {1}
var c:real;
begin
  while abs(b-a)>2*e do begin
    c:=(a+b)/2;                                  {2}
    if (3*a*a*a+4)*(3*c*c*c+4)<=0 {3}
    then b:=c else a:=c end;
    F:=(a+b)/2;
  end;
begin
  a:= -2; b:= -1;

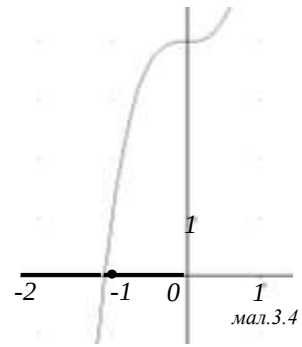
```

```

WriteLn('e?');ReadLn(e);
WriteLn('x=', F(e,a,b):8:6);      {4}
end.

```

Програма видає наближені значення розв'язку рівняння $3x^3+4=0$ при $\varepsilon=0,1$ $x=-1,0625$, при $\varepsilon=0,0001$ $x=-1,100647$. При зменшенні ε значення розв'язку стає більш точним. У першому випадку точність не перевищує 0,1. У другому випадку точність досягає 0,0001. Щоб побачити процес уточнення, розглянемо програму *SpecDec**.



```

program SpecDec*;                  {f(x)=3x^3+4}
var i:integer;
    a,b,c,x,e:real;
{опис функції F взяти з програми SpecDec};
begin
    a:=-2; b:=-1;
    e:=abs(b-a)/2;                 {1}
    for i:=1 to 12 do begin        {2}
        WriteLn('x=',F(e,a,b):12:10);
        e:=e/10                    end;    {3}
    end.

```

У ній ε змінне, воно поступово зменшується, починаючи з $abs(b-a)/2$ {1}, тобто точність не перевищує 1. Кожне наступне значення ε менше попереднього в 10 разів {2}, тобто точність зростає на порядок, стаючи рівною 0,1; 0,01; 0,001 і т.д. Це видно із таблиці 3.1:

						таблиця 3.1
ε	0,5 (точ. 1)	0,05 (точ. 0,1)	0,005 (точ. 0,01)	0,0005	0,00005	0,000005
x	-1,5	-1,09375	-1,09765625	-1,1010742188	-1,1006164551	-1,1006431580
ε	0,0000005	0,00000005	0,000000005	0,0000000005	0,000000000005	0,00000000000005
x	-1,1006426811	-1,1006424129	-1,1006424166	-1,1006424162	-1,1006424163	-1,1006424163

Два останні значення x дорівнюють -1,1006424163, що означає досягнення точності, дозволеної параметром $l2$ циклу {2} та форматом виводу {3}. Для досягнення вищої точності слід збільшити ці величини. Але уточнення розв'язку програмою *SpecDec** можливо лише в межах типу *real* (15-16 значущих цифр).

Недоліком наведених програм є те, що кількість цифр результату не залежить від встановленої точності. Програма *SpecDec*** самостійно встановлює кількість цифр виводу, для цього вона містить додаткову функцію *NumMark* (див {!}).

```

program SpecDec**;                  {f(x)=3x^3+4}
var i,n:integer;
    a,b,c,x,e:real;

```



```

procedure InpDat(var a,b,e:real);
begin
  WriteLn('a,b?');ReadLn(a,b);
  WriteLn('e?');ReadLn(e);
end;{end InpDat}
function NumMark(e:real):integer;
var i:byte;eps:real;
begin
  i:=0;eps:=e;
  while eps<=1 do      begin
    eps:=eps*10;i:=i+1 end;
  NumMark:=i
end;{end NumMark}
function SpecRoot(e,a,b,c:real):real;
begin
  while b-a>2*e do      begin
    c:=(a+b)/2;
    if (3*a*a*a+4)*(3*c*c*c+4)<=0
      then b:=c else a:=c end;
    SpecRoot:=c;
  end;{end SpecRoot}
begin
  InpDat(a,b,e);
  n:=NumMark(e);
  WriteLn('x=', SpecRoot(e,a,b,c):15:n) {!}
end.

```

Приклади роботи цієї програми для різних значень ε :

$\varepsilon=0,01$, $x=-1,102$; $\varepsilon=0,0001$, $x=-1,10065$; $\varepsilon=0,000001$, $x=-1,1006422$.

? **Питання до практичної роботи:**

3.1 Яка роль ε при уточненні розв'язку рівняння?

3.2 Що слід змінити у програмі **SpecDec****, щоб вона уточнювала розв'язок іншого рівняння?

3.3 Чи можна уточнювати розв'язки рівняння на проміжку $[a,b]$, якщо на ньому графік відповідної функції має розриви?

✎ Завдання 3.1 Написати програму, яка визначає усі проміжки, що містять розв'язки рівняння $p(x)=0$, де $p(x)$ – многочлен.

Завдання 3.2 Записати програму **SpecDec**** у вигляді функції.

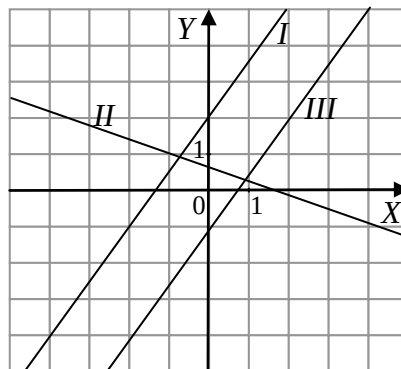
Завдання 3.3* Написати програму, яка з допомогою функції **SpecDec**** визначає з указаною точністю усі розв'язки рівняння $p(x)=0$, де $p(x)$ – многочлен.

Практична робота 4. РОЗВ'ЯЗУВАННЯ СИСТЕМИ ДВОХ (ТРЬОХ) ЛІНІЙНИХ РІВНЯНЬ З ДВОМА (ТРЬОМА) НЕВІДОМИМИ.

Нехай дано систему двох лінійних рівнянь з двома невідомими:

(1)

$$\begin{cases} a_1x + b_1y = c_1, \\ a_2x + b_2y = c_2 \end{cases}$$



мал. 4.1

Як відомо, графіком кожного з них є пряма лінія. Дві прямі можуть мати лише три варіанти взаємного розміщення (див. мал.1): бути паралельними, співпадати, або перетинатись в одній точці. Отже система (1) може або не мати розв'язків (коли $a_1/a_2 = b_1/b_2 \neq c_1/c_2$), або мати безліч розв'язків (коли $a_1/a_2 = b_1/b_2 = c_1/c_2$), або мати єдиний розв'язок (коли $a_1/a_2 \neq b_1/b_2$).

Ці співвідношення можна покласти в основу алгоритму, але тоді необхідні додаткові формули для визначення розв'язків. Тому зручніше скористатись *правилом Крамера*. Але для цього спочатку введемо позначення визначника системи двох лінійних рівнянь з двома змінними Δ та визначників Δ_x і Δ_y :

$$\Delta = \begin{vmatrix} a_1 & b_1 \\ a_2 & b_2 \end{vmatrix} = a_1b_2 - a_2b_1; \quad \Delta_x = \begin{vmatrix} c_1 & b_1 \\ c_2 & b_2 \end{vmatrix} = c_1b_2 - c_2b_1; \quad \Delta_y = \begin{vmatrix} a_1 & c_1 \\ a_2 & c_2 \end{vmatrix} = a_1c_2 - a_2c_1; \quad (2)$$

Формули Крамера мають вигляд: $x = \frac{\Delta_x}{\Delta}, y = \frac{\Delta_y}{\Delta}$ (3)

Тоді програма для розв'язування системи двох лінійних рівнянь з двома невідомими методом Крамера матиме вигляд:

```
program SystLinEquat;
  var i,j:integer;d,dx,dy,x,y:real;
      a:array[1..2,1..3]of real;
begin
  for i:=1 to 2 do
    for j:=1 to 3 do begin
      WriteLn('a['i,',',j,']?'); ReadLn(a[i,j]) end;
    d:=a[1,1]*a[2,2]-a[2,1]*a[1,2];
    dx:=a[1,3]*a[2,2]-a[2,3]*a[1,2];
    dy:=a[1,1]*a[2,3]-a[2,1]*a[1,3];
    if d<>0 {1}
    then begin {2}
      x:=dx/d; y:=dy/d; {2}
      WriteLn('x=',x); {2}
      WriteLn('y=',y) end {2}
    else if dx=dy {3}
      then WriteLn('розв. безліч')
      else WriteLn('розв. нема');
    end.
```

У програмі визначники позначено літерою "d". Команди {2} можна замінити на {4}: WriteLn('x=',dx,',',d,' y=',dy,',',d) {4}. Це не тільки скоротить

кількість команд, а й дасть змогу завжди одержувати точні розв'язки. У *SystLinEquat* {1} виконує перехід до випадку, коли розв'язки існують ($d \neq 0$). Відповідно {3} передбачає, що $d=0$. Згідно правила Крамера, якщо $d=d_x=d_y=0$, розв'язків безліч, інакше – жодного. Перевірку $d_x=d_y=0$ можна замінити на $d_x=d_y$ (перевірте самостійно).

(4)

Описаним методом можна розв'язувати й системи трьох лінійних рівнянь з трьома невідомими:

Для цього замість формул (2) – (3) треба взяти формули (5) – (7):

$$\Delta = \begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1 b_2 c_3 + b_1 c_2 a_3 + c_1 a_2 b_3 - c_1 b_2 a_3 - a_1 c_2 b_3 - b_1 a_2 c_3 \quad (5)$$

$$\Delta_x = \begin{vmatrix} d_1 & b_1 & c_1 \\ d_2 & b_2 & c_2 \\ d_3 & b_3 & c_3 \end{vmatrix}; \quad \Delta_y = \begin{vmatrix} a_1 & d_1 & c_1 \\ a_2 & d_2 & c_2 \\ a_3 & d_3 & c_3 \end{vmatrix}; \quad \Delta_z = \begin{vmatrix} a_1 & b_1 & d_1 \\ a_2 & b_2 & d_2 \\ a_3 & b_3 & d_3 \end{vmatrix} \quad (6)$$

$$x = \frac{\Delta_x}{\Delta}, \quad y = \frac{\Delta_y}{\Delta}, \quad z = \frac{\Delta_z}{\Delta} \quad (7)$$

Пропонуємо формули для обчислення Δ_x , Δ_y і Δ_z написати та перевірити самостійно, по зразку (5). Можна замість формули (5) використовувати формулу (7):

$$\begin{vmatrix} a_1 & b_1 & c_1 \\ a_2 & b_2 & c_2 \\ a_3 & b_3 & c_3 \end{vmatrix} = a_1 \begin{vmatrix} b_2 & c_2 \\ b_3 & c_3 \end{vmatrix} - b_1 \begin{vmatrix} a_2 & c_2 \\ a_3 & c_3 \end{vmatrix} + c_1 \begin{vmatrix} a_2 & b_2 \\ a_3 & b_3 \end{vmatrix} \quad (8)$$

? Питання до практичної роботи:

4.1 Скільки розв'язків може мати система трьох лінійних рівнянь з трьома невідомими?

4.2 Яка з формул (5) чи (7) краща для використання при обчисленні "вручну"?

4.3 Яка з формул (5) чи (7) краща для використання у програмі?

✎ Завдання 4.1 Обчислити "вручну" визначник $\begin{vmatrix} 2 & -8 \\ 4 & 5 \end{vmatrix}$.

Завдання 4.2 Скласти систему рівнянь, відповідну визначнику завдання 4.1, взявши вільні члени довільно.

Завдання 4.3 Розв'язати методом Крамера систему завдання 4.2.

Завдання 4.4 Обчислити "вручну" визначник $\begin{vmatrix} -4 & 7 & 3 \\ 0 & 2 & 0 \\ -8 & -3 & 6 \end{vmatrix}$.

Завдання 4.5 Скласти систему рівнянь, відповідну визначнику завдання 4.4, взявши вільні члени довільно.

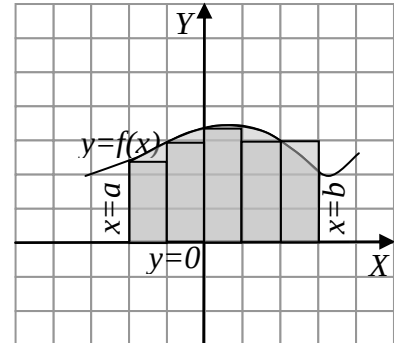
Завдання 4.6 Розв'язати методом Крамера систему завдання 4.5.

Завдання 4.7* Змінити програму *SystLinEquat* так, щоб для вводу і виводу використовувались текстові файли.

Завдання 4.8* Написати програму *SystLinEquat3*, яка визначає розв'язки системи трьох лінійних рівнянь з трьома невідомими методом Крамера.

Практична робота 5. НАБЛИЖЕНЕ ОБЧИСЛЕННЯ ПЛОЩІ КРИВОЛІНІЙНОЇ ТРАПЕЦІЇ.

❶ Криволінійною трапецією називається фігура на координатній площині, обмежена віссю OX , прямими $x=a$, $x=b$ та графіком деякої функції $y=f(x)$.

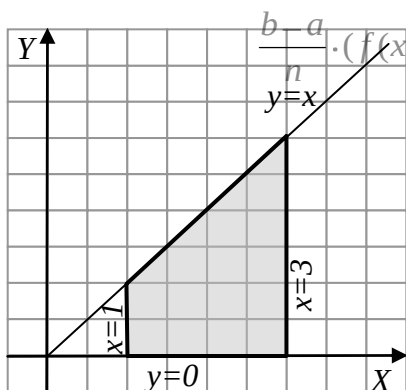


мал. 5.1

Існує багато способів наближеного обчислення площі криволінійної трапеції. Один із них (див. мал.) зводиться до обчислення площі ступінчатої фігури, що складається із n дотичних прямокутників однакової ширини $h=(b-a)/n$ і висотою $f(x_i)$, де x_i – абсциса лівого краю i -го прямокутника. Площа ступінчатої фігури буде наближатись до площі криволінійної трапеції із збільшенням n – кількості “сходинок”, тобто буде наближеним значенням площі криволінійної трапеції з великою точністю, якою можна, як і в практичній роботі 3, управляти. Формула обчислення площі має вигляд:

(1)

$$S = \frac{b-a}{n} \cdot f(x_0) + \frac{b-a}{n} \cdot f(x_1) + \dots + \frac{b-a}{n} \cdot f(x_{n-1}) =$$



мал. 5.2

П
ро-
гр
ама обчислення площі може мати такий вигляд:

```
program Area;
  var i,n:integer;
      a,b,h,x,s:real;
begin
```

```
  WriteLn('a?');ReadLn(a);
```

```
  WriteLn('b?');ReadLn(b);
```

```
  WriteLn('n?');ReadLn(n);
  h:=(b-a)/n;           {1}
  s:=0;
  x:=a;
  for i:=1 to n do begin
    s:=s+h*sqrt(x);      {2}
    x:=x+h               end; {3}
  WriteLn('s=',s:12:6);
end.
```

Величина h називається *кроком*, вона дорівнює $(b-a)/n$ {1}. Початкове значення аргументу x дорівнює a , кожне наступне $x+h$ {3}. У програмі взято функцію $y=\sqrt{x}$, тому змінною частиною програми є рядок {2}. Наведемо залежність значення площі від вибраного кроку. Для цього візьмемо функцію $y=x$, $a=1$, $b=3$. З малюнка 5.2 видно, що площа виділеної криволінійної трапеції дорівнює 4 кв.од.

таблиця 5.1

n	4	10	15	20	30	40	50	100	200	1000	2000
s	3,5	3,8	3,8(6)	3,9	3,9(3)	3,95	3,96	3,98	3,99	3,998	3,999

? Питання до практичної роботи:

- 5.1 Як позначиться на результаті заміна у формулі (1) значень $x_0, x_1, x_2, \dots, x_{n-1}$ на $x_1, x_2, x_3, \dots, x_{n-1}, x_n$?
- 5.2 Чи можна обчислити площу криволінійної трапеції на проміжку $[a,b]$, якщо на ньому графік відповідної функції має розриви?
- 5.3 Який вигляд матиме криволінійна трапеція для функції $y=x$ при $a=-1, b=3$?

✎ Завдання 5.1 Обчислити “вручну” за формулою (1) площу фігури, утвореної функцією $y=x^2$ на відрізку $[1;3]$ для $n=5$.

Завдання 5.2 Записати програму *Area* у вигляді функції.

Завдання 5.3* Написати програму, яка з допомогою функції *Area* виводить на екран таблицю залежності значення площі від значення n .

Завдання 5.4* Написати програму, яка з допомогою функції *Area* визначає наближене значення площі із указаною точністю ϵ (див. практичну роботу 3).

Практична робота 6. РОЗПОДІЛ ТЕМПЕРАТУР В ОДНОРІДНОМУ СТЕРЖНІ.

Задача 1: На кінцях однорідного теплопровідного стержня підтримується температура t_1 і t_2 , а по всій довжині стержень теплоізований. Визначити, яка температура установиться у внутрішніх частинах стержня.

Розіб'ємо стержень на достатньо малі частини, температуру в яких можна вважати однаковою. Очевидно, що у кожній з таких частин температура у кожен момент часу визначатиметься за формулою: $t=(t_{\text{left}}+t_{\text{right}})/2$. Процес зміни температур матиме вигляд:

таблиця 6.1

t_{left}	t_1	t_2	t_3	t_4	t_5	t_{right}
10	0	0	0	0	0	50
10	5	0	0	0	25	50
10	5	2,5	0	12,5	25	50
10	6,25	2,5	7,5	12,5	31,25	50
10	6,25	6,875	7,5	19,375	31,25	50
10	8,4375	6,875	13,125	19,375	34,6875	50
10	8,4375	10,78125	13,125	23,90625	34,6875	50
10	10,390625	10,78125	17,34375	23,90625	36,953125	50
10	10,390625	13,8671875	17,34375	27,1484375	36,953125	50

10	11,9335938	13,8671875	20,5078125	27,1484375	38,5742188	50
10	11,9335938	16,2207031	20,5078125	29,5410156	38,5742188	50
10	13,1103516	16,2207031	22,8808594	29,5410156	39,7705078	50
10	13,1103516	17,9956055	22,8808594	31,3256836	39,7705078	50
10	13,9978027	17,9956055	24,6606445	31,3256836	40,6628418	50
10	13,9978027	19,3292236	24,6606445	32,6617432	40,6628418	50
10	14,6646118	19,3292236	25,9954834	32,6617432	41,3308716	50
10	14,6646118	20,3300476	25,9954834	33,6631775	41,3308716	50
10	15,1650238	20,3300476	26,9966125	33,6631775	41,8315887	50
10	15,1650238	21,0808182	26,9966125	34,4141006	41,8315887	50
10	15,5404091	21,0808182	27,7474594	34,4141006	42,2070503	50
10	15,5404091	21,6439342	27,7474594	34,9772549	42,2070503	50

У таблиці 6.1 показано 20 кроків. Останні рядки свідчать, що температура в частинках стержня встановлюється сталою з досить великою точністю (виділено шрифтом ***Bold***).

Програма визначення температур у частинках стержня має вигляд:

```

program Core;
    const n=5;                                {1}
    type A=array[0..n+1] of real;
    var i,k:integer;
        tL,tR,t0,eps,e:real;
        T,Th:A;f:char;
begin
    tL:=10;tR:=20;t0:=-1000;eps:=0.001; {2}
    k:=0;e:=eps;
    repeat
        k:=k+1;e:=e*10 until not(e<1);
        Th[0]:=tL;Th[n+1]:=tR;
        for i:=1 to n do Th[i]:=t0;
        repeat
            for i:=1 to n do begin
                T[i]:=(Th[i-1]+Th[i+1])/2;
                Write(T[i]:k+1:k,' ') end;
            f:='y';WriteLn;
            for i:=1 to n do
                if abs(T[i]-Th[i])>=eps then f:='n';
            for i:=1 to n do Th[i]:=T[i];
        until f='y';
    end.

```

Ця програма дозволяє встановити n – кількість частинок стержня {1} та eps – точність, з якою треба визначити остаточні температури в частинках {2}. Для $n = 5$ та $eps = 0,001$ два останні рядки таблиці мають вигляд:

таблиця 6.2						
t_{left}	t_1	t_2	t_3	t_4	t_5	t_{right}
10	116665	13,330	14,997	16,663	18,332	20
10	116665	13,331	14,997	16,664	18,332	20

Це свідчить, що процес теплообміну зупинився з точністю до 0,001.

Задача 2: На кінцях однорідної теплопровідної квадратної пластини підтримується температура t_1 , t_2 , t_3 і t_4 , а по всій довжині пластини теплоізована. Визначити, яка температура установиться у внутрішніх частинах пластини.

? Питання до практичної роботи:

6.1 Яка формула відображає теплообмін у пластині (задача 2)?

6.2 Чи закінчиться теплообмін у стержні та пластині (задачі 1-2) взагалі?

✎ **Завдання 6.1** Змінити програму *Core*, щоб вона визначала час теплообміну до зупинки з указаною точністю, вважаючи, що кожен крок виконується за одну секунду.

Завдання 6.2* Написати програму, що визначає теплообмін в частинах пластини (задача 2) з указаною точністю.

ДОДАТОК 3

ПРО МОТИВАЦІЮ НАВЧАЛЬНОЇ ДІЯЛЬНОСТІ

ТРЕБА – ХОЧУ – ЗМОЖУ

Інколи дивуєшся, скільки красивих, і, що найголовніше, в цілому вірних фраз вигадали педагоги про своїх вихованців. Досить згадати: «Дитина – це не посудина, яку треба наповнити, а факел, який треба запалити», «Немає поганих учнів, є погані вчителі», «Ставлячи двійку, вчитель розписується в своїй безпорадності» і т.д. Але ці фрази вірні аж до банальності і надто часто вживані. Як наслідок: учителі їх проголошують за звичкою, батьки на батьківських зборах пропускають повз вуха. А діти? Але при чому тут діти? Це ж не для них, а про них, без їх відома та згоди. Тому, як наслідок, позитивних результатів нема. Навпаки, час від часу доводиться констатувати, що з освітою та вихованням все гірше. Чому? Важко відповісти на це вічне запитання. Але саме тому, що воно вічне, нам слід знову і знову невтомно шукати відповідь, яка частково щойно вже висловлена: «не для них, а про них, без їх відома та згоди». Справді, подумаймо, якби нас з вами хтось «запалював», у певному розумінні, а нам треба було «горіти смолоскипом», як би ми подивились на таку перспективу? А чи не лукавимо ми, погоджуючись, що всі учні, без винятку, тільки білі та пухнасті? Погодьмось, тут, як про поганих дружин, коли усі дівчата хороші. І на сонці є темні п'ятна. І діти мають недоліки, адже вони також люди, причому ще не сильні, не загартовані, а про людей вже 2000 років тому сказано: «Киньте в мене каменем, хто сам не грішний». А «геніальна» думка про безпорадність учителів, які ставлять двійки? Адже виходить абсурд: з одного боку є критерії оцінювання, зокрема початковий рівень знань, з іншого, ставити двійки – майже педагогічний злочин. Ставити оцінки, в тому числі й двійки – це складова процесу навчання та виховання. Інша справа, що необхідна конструктивна робота по мотивації навчальної діяльності учнів та використання прогресивних технологій навчання учителями. Причина канонізації подібних прописних істин – це наслідок нашого горя від власного розуму, як сказав класик (О.С. Грибоєдов).

Отже, «лукаво не мудруючи», спустимось з ліричних хмаринок на грішну землю і подивимось на навчально-виховний процес з позиції суворой реальності, без зайвої гри в гуманізацію, демократизацію, інновації, реформації і т.д. А реальність така: *сучасні учні – це продукт сучасного суспільства*, яке дуже швидко змінюється. Не дивно, що ми, як представники чи не найконсервативнішої професії, не встигаємо за плином часу, неадекватно реагуємо на зміни, замість того без причини сіпаємось, реформуємо, витісняючи з уроку письмове мовлення, замінюючи його усним, замість логічного мислення та аналізу надмірно унаочнюємо, змушуємо учнів пасивно спостерігати доречні і недоречні комп'ютерні презентації, без яких тепер урок уже не урок. Учні паралельно встигають слідкувати за подіями у власному мобільному телефоні, при нагоді просяться вийти, щоб поговорити з таким же товаришем про щось інше. Ми збіднили форми самостійної роботи на уроці, тепер учні вибирають (вірніше – вгадують) правильну відповідь серед чотирьох запропонованих, заповнюють так звані «робочі зошити» тощо. І ми в цьому не винні. Адже все йде по вертикалі, нам треба час від часу атестуватись, треба, щоб атестаційна комісія виставила більше «плюсів» і менше «мінусів». Виходить суєта суєт і нескінченна та малоефективна гонитва за результатами, а якщо насправді й чесно – за показниками. Даєш ІНТЕЛ, бо по вертикалі звітують і треба не пасти задніх. То нічого, що в комп'ютерних технологіях основна маса учителів практично там, де була й 10 років тому, але ж сертифікати Інтел мають уже більшість учителів. Як влучно говорив А. Райкін, замість «якості» даємо «количество». Зате сучасно! І ось ми вже повторюємо за кимось відомим, що учні – це співавтори уроку. Не схоже, що, як Остапа [Бендера] нас понесло? Пам'ятаю, колись у нашій школі були представники педагогічної школи вчителя-новатора Ільїна. В кінці показового, справді майстерного, уроку методист подякував учням за урок. Це було тоді справді дуже доречно, адже учні дозволили вчителеві сказати все, що він хотів, навчити їх всього, що він знав. Знову запитання: а чому? І тепер, нарешті можна перейти до головного сьогоднішнього питання – до **мотивації навчання**. Тоді, понад двадцять років тому на незабутньому для мене уроці в учнів була мотивація до пізнання, якою учитель геніально скористався.

Про головне слід говорити коротко, але про мотивацію навчання, її сучасний стан та пошуки шляхів покращення можна написати не одну дисертацію, адже тема дуже широка й глибока, а на сучасному фактичному матеріалі практично не досліджена.

Існують різні погляди на мотивацію поведінки, які складають науку «Теорія мотивації». Можна назвати: теорії потреб Маслоу та Альджерфера, теорію Х іа Y-мотивацій Д. МакГрегора і т.д. Наприклад, по МакГрегору теорія Х передбачає, що працівники (учні) спочатку ліниві і будуть по можливості уникати роботи. Через це працівники повинні бути під пильним наглядом, для чого розробляються комплексні системи контролю. Теорія Y припускає, що працівники (учні) можуть бути амбітними, мати внутрішні стимули, прагнути взяти на себе більше відповідальності і здійснювати самоконтроль і самоврядування, напрямку. Вважається, що вони отримують задо-

волення від своїх обов'язків, пов'язаних як з розумовим, так і фізичною працею, відчують бажання проявляти творче і прогресивне мислення у виробництві, якщо випаде можливість. У наукових теоріях проведена спеціальна класифікація мотивацій Шольца.

Але недоліком усіх канонічних теорій є їх відірваність від життя, кабінетність, неможливість у чистому вигляді застосувати на практиці, тому, безперечно, враховуючи основні положення цих теорій, слід шукати практичні методики управління мотиваційною діяльністю учнів, виходячи з реальності. Отже спробуємо провести дуже короткий аналіз.

Перш за все приймемо, що **МОТИВИ НАВЧАННЯ** - це створення у школі умов для появи внутрішніх спонукань до навчання, усвідомлення їх учнями і подальшим саморозвитком власної мотиваційної сфери.

В основі мотивації лежать притаманні людям потреби, бажання, поведінка, емоції, почуття, біль, страх тощо, які разом утворюють мотиваційний комплекс.

Звідси впливають мотиви:

- **обов'язку і відповідальності** перед суспільством, групою, окремими людьми;
- **самовизначення і самовдосконалення**;
- **прагнення отримати схвалення** інших людей;
- **прагнення отримати високий соціальний статус** (престижна мотивація). При відсутності інтересу до діяльності (процесуально-змістовна мотивація) існує прагнення до тих зовнішніх атрибутів, які може принести діяльність, - до відмінних оцінок, до отримання диплома, до слави в майбутньому;
- **уникнення неприємностей і покарання** (негативна мотивація) - спонукання, що викликає усвідомлення деяких неприємностей, незручностей, які можуть виникнути в разі невиконання діяльності.
- **прагнення досягнень** високих результатів та майстерності у діяльності; воно проявляється у виборі складних завдань і прагненні їх виконати. Успіхи в будь-якій діяльності залежать не тільки від здібностей, навичок, знань, а й від мотивації досягнення. Людина з високим рівнем мотивації досягнення, прагнучи отримати вагомий результат, наполегливо працює задля досягнення поставлених цілей.

Поведінка учнів залежить від стимуляції, мотивації та впливів. Стимуляція завжди виходить тільки ззовні, може бути як позитивною, так і негативною, але тривалого ефекту для формування особистості ніколи не приносить. Її можна часто використовувати в тактичних цілях. Впливи – це також зовнішні чинники, їх роль для формування особистості може бути і позитивною і негативною. Але на відміну від стимуляції впливи можуть бути тривалими. До впливів слід ставитись дуже обережно, адже вони завжди бувають спонтанними й стихійними. До негативних впливів, в першу чергу таких, що приходять через засоби масової інформації, телебачення, інтернет слід придивитись стійкий імунітет.

МОТИВАЦІЯ НАВЧАННЯ – це самостимуляція навчальної діяльності. Перш за все вона може бути **внутрішньою** (*інтринсивна* мотивація, *автомотивація*) та **зовнішньою** (*екстринсивна* або *інтерактивна*). Роль внутрішньої мотивації виняткова, вона відіграє головну роль, бо саме через неї одержується кінцевий результат. Крім того, автомотивація стійка, адже в її основі лежить переконання. Застосування зовнішньої стимуляції навчання хоч і необхідне, але домінувати може лише на початкових етапах, в ході навчально-виховного процесу її слід застосовувати дозовано, лише при нагальній потребі.

Мотивація навчальної діяльності повинна бути не штучною, а переконливою і зрозумілою.

Треба також згодитись, що об'єктивно існує **рівень мотивації**, тобто її *величина* або *вимір*, адже можна говорити про велику і незначну мотивацію. Очевидно, що слід ввести також поняття **негативної мотивації**, іншими словами – *антимотивації*, яка може бути як спеціальною, свідомою, а й невимушеною чи прихованою. Це означає, що невміла мотивація, або така, що застосовується незграбно, непродумано, може давати протилежний ефект, який може навіть перевершити допустимі межі.

Слід розрізняти *індивідуальну та колективну, локальну і глобальну, тактичну й стратегічну, тривалу і нестійку, дослідницька та ігрова* мотивації. Мотивація навчальної діяльності не може бути окремим засобом в руках педагога, вона завжди поєднується з усім комплексом навчальної й педагогічної роботи, часто виступає в ролі незначного прихованого чи допоміжного, а насправді головного інструменту навчально-виховного процесу.

Ні в якому разі на мотивацію навчання не можна дивитись, як на інноваційний засіб, або на новий винайдений велосипед, вона була завжди присутня в навчально-виховному процесі, із зміною суспільства необхідно лише вчасно видозмінювати форми та прояви мотивації. Приміром, 15-20 років тому актуальною була необхідність зацікавлювати предметом через цікаві задачі, приклади з історії науки тощо.

Головне завдання школи, як було раніше і завжди, – це висока **ЯКІСТЬ ОСВІТИ** (за розумними, а не заморськими мірками). Відомо, що освіта – це те, що залишилось коли все вивчене забуте. Першим принципом мотивації навчання повинна бути **СТИСЛІСТЬ**. Краще вчити небагато, але так, щоб забувалось з вивченого не 80-90%, а не більше 10-15%. Тут слід згадати перше *антимотиваційне* правило, якого дотримуються ті, хто доводить до нас програми та підручники. Зміст одного уроку згідно програми ніяк не втиснути в рядок, де в класному журналі записується тема. Відповідно сучасним підручникам, щоразу учні мусять опрацьовувати 12-15 сторінок тексту підручника, тобто щодня прочитати й засвоїти 80-100 друкованих сторінок із запитаннями та завданнями. Величезне перевантаження, що заважає мотивації навчання! Нормальний учень у таких умовах зовсім не буде відкривати підручник.

Якість освіти – результат **ЯКОСТІ НАВЧАННЯ**. Отже слід думати про високу якість навчальної роботи. В минулому навчальному році я виступав з ініціативою щодо поліпшення якості навчання через об'єктивні й незалежні методи визначення якості знань з допомогою інформаційно-комунікаційних технологій, що дозволяє удосконалити прогнозування й планування якісних показників навчання. При цьому було зауважено, що «у наш час чи не найважливішим чинником якості навчання й освіти в цілому є **мотивація навчання**». Але розгляд цієї теми тоді свідомо було залишено поза межами дослідження якості навчання. Мотивацію навчальної діяльності для якості навчання й освіти слід розглядати, як вітаміни, які не мають енергетичної цінності, не є «будівельним матеріалом» для організму, а виконують функцію каталізатора обмінних процесів. Отже **мотивація навчання покликана активізувати й прискорювати навчально-виховний процес**.

Справжня **мотивація навчання** має три основні складові, тобто мотиваційні принципи:

- а) **ТРЕБА** (цей мотив навчання несправедливо вважається застарілим, адже впливає з марксистського визначення категорії свободи, як усвідомленої необхідності);
- б) **ХОЧУ** (на жаль сьогодні цей мотив поведінки, який для навчання важливий, але не головний, помилково вважається пріоритетним).
- в) **МОЖУ** (є наслідком мотиву «треба»);

Позитивна й здорова мотивація навчання у сьогоднішніх учнів практично відсутня. Її можна сформулювати так: хоч канаву рити від Щасливого до Києва, якби не сидіти на уроках. Дійсно, «на ура» сприймаються, наприклад, репетиція і проведення святкового концерту до Дня учителя, але тільки за рахунок уроків. Готуватись до олімпіад – теж переважно за рахунок інших уроків. При опитуваннях учні висловлюються за те, щоб учителі не давали домашніх завдань, не питали на уроках. Важко знайти учня, який хоче брати участь у предметних гуртках, готуватись до олімпіад. Але чому? Тільки чесно давши відповідь на це запитання можна приймати рішення щодо підвищення мотивації учнів до навчання.

Думаючи над темою мотивації, інколи корисно на уроці організувати експериментальний майданчик. Ні для кого не секрет, що мотивація до навчання учнів старшої ланки найнижча. Тому інколи доводиться третину уроку присвятити роздумам про подальші перспективи випускників. Приводом може стати наприклад те, що учні масово не готуються до заліку. Враховуючи те, що одинадятикласники фактично дорослі люди, випускники, розмова повинна бути відвертою і принциповою, із прихованим застосуванням психологічних методів. Учня можна запропонувати знайти відповіді на незручні питання про розмови з батьками, про перспективи закінчення школи та вступу, про планування дорослого життя, а головне: для чого тратити дорогоцінний, неповторний час молодого людини на школу, якщо не прикласти жодних зусиль для одержання якісної освіти. Щоб спровокувати відвертість, бесіда може проводитись у формі опитування. Корисно застосовувати навіть прийоми «психологічної атаки», коли, не називаючи прізвищ,

можна навести приклади спостережень. Учні поступово переконуються, що навчальна діяльність потрібна не вчителям, а їм самим. Поступово частина учнів всерйоз задумується над своїми проблемами і перспективами. Тоді результатів доведеться чекати не довго. На протязі тижня частина учнів стане підходити для консультації чи виправлення оцінки. Серед них будуть і такі, хто раніше ніколи з цією метою до вчителя не підходив через байдужість або невіру в свої сили.

Ще раз окремо слід задуматись над мотивацією учнів до творчої навчальної діяльності і поглибленого вивчення основ наук. Мається на увазі участь в предметних олімпіадах, конкурсах, в секціях Малої академії наук. Не секрет, що багато учнів, хоч і приймають участь у такій роботі, але деже нехотя, без достатньої мотивації. У той же час, десяток років тому, така мотивація була дуже значною. Яка причина збайдужіння наших учнів. Безперечно – перевантаження. Абсолютно неприпустимою є участь одного учня в II-III етапах трьох а то й чотирьох предметних олімпіад і в одній-двох секціях МАН. Але не це головне. Першим мотивом для поглибленого вивчення певного предмета може бути тільки достатньо високий рівень базових знань з нього. Колись цей рівень справді був на висоті. А сьогодні, на фоні базового рівня знань, що ледве досягає «плінтуса» ми спонукаємо учнів не тільки брати участь в олімпіадах, а й масово перемагати. Це непосильне завдання, тому учні так неохоче працюють у цій сфері. Хтось незгоден? Тоді давайте відмінімо в цілому негативну практику підготовки до олімпіад шляхом «вільного відвідування», коли учні на протязі не днів, а тижнів пропускають уроки, готуючись до якоїсь олімпіади. Чи багато дітей будуть з нами працювати? Принаймні не більше 10% від теперішньої кількості. А хіба може бути мотивом до навчання, коли за участь в олімпіадах учням ставлять «призові» бали в журналах?

Думаючи про мотивацію навчання, слід згадати Основний закон країни. Керуючись конституційним правом учня на навчання, необхідно зробити за правило ні під яким приводом не звільняти та знімати учнів з уроків. **Навчання повинне бути нормою, а не винятком у житті школярів!** Учні також мають конституційне право на відпочинок, зокрема в канікулярний час. Ігнорування згаданих прав приводить до значного зниження мотивації навчання. Вважаю недопустимими довгострокові звільнення значної кількості учнів від навчання, що мало місце у вересні за вказівкою «зверху» для підготовки та участі в турнірі Майданс та брейн-ринг на приз губернатора. На серпневій конференції Міністр освіти, науки, молоді та спорту Д. В. Табачник сказав, що уже охоплено понад 60% учнів всілякими конкурсами, олімпіадами та змаганнями. А коли ж учням учитись? Звідки взяти таким чином перевантаженим дітям позитивну мотивацію до навчання? Очевидно рапорт Міністра став меседжем по вертикалі, бо тільки з інформатики у цьому році замість однієї олімпіади стало дві та ще й з конкурсом Бобер у придачу.

Очевидно має місце перекручення зверху, якщо не повне ігнорування чиновницьким корпусом мотиваційних принципів навчання: **ТРЕБА–ХОЧУ–МОЖУ**. В таких умовах у школі практично не можливо забезпечити

виконання учнями **ОБОВ'ЯЗКУ НАВЧАТИСЬ**. Що робити в таких умовах учителеві? Єдине – справою доводити, що наша професія творча. А **ТВОРЧІСТЬ** не **МОЖЕ БУТИ**, за вказівкою, з-під палки, вона може бути **ТІЛЬКИ ВІЛЬНОЮ**. То ж маймо мужність бути вільними, професійно робити свою справу, не дозволяючи втручатись у цей процес ні модним шоу-подібним прийомам, ні чиновникам, які вносять в навчально-виховний процес хаос. І **НЕ ВЧИТИ** учнів, а допомагати їм **УЧИТИСЬ**. Тільки так можна досягти позитивної мотивації в навчанні.

Для формування позитивних характеристик особистості і досягнення успіхів повинна бути стійка мотивація. Ефективна мотивація буває тільки внутрішньою, вона повинна стати важливою характеристикою особистості. Мотивація повинна бути свідомою, ґрунтуватись на морально-етичних та суспільно-світоглядних позиціях і переконаннях.

Перед школою стоять питання, як будувати навчально-виховну роботу у сучасному, ворожому для інтелектуальної та виховної сфери суспільстві, коли засоби масмедіа підпорядковуються так званій редакційній політиці (а насправді виконують замовлення власників) і в той же час повністю залежну від них журналістику називають «четвертою владою», коли проходить планомірне *розщеплення свідомості*, культивується і насаджується *мозаїчна культура*, як незаперечний позитив подається епатажність та піар, це напряму пов'язується з філософією успішності.

Чому у Великобританії на тлі дуже низького рівня освіти існують елітні (так звані публічні школи) зразка Кембріджського та Оксфордського осередків? Чому в нас буквально нав'язується все найгірше з європейської освіти, а елітні навчальні заклади, що тепер виникають, як гриби після дощу, не стають аналогами елітних навчальних закладів на Заході?

Очевидно, нам не слід переймати чужі освітні постулати, а формувати стійку мотивацію до навчання, естетичності та моральності. Слід пам'ятати, що з народження в людини присутня розвинена емоційна і лише у задатках інтелектуальна сфера, а сенсом життя повинні стати розвиток інтелекту та навичок його домінування над емоційністю.

ВПРОВАДЖЕННЯ БІНАРНИХ УРОКІВ У ПРОЦЕС ВИКЛАДАННЯ ІНФОРМАТИКИ.

При викладанні основ інформатики неможливо обійтись без міжпредметних зв'язків, однією з найефективніших форм яких є бінарні уроки. Поняття бінарного уроку в сучасному розумінні можна сформулювати так : *бінарними є уроки, що поєднують два предмети в одній темі.*

Наприклад, можна говорити про бінарні уроки *інформатика – математика*. Безперечно, на кожному бінарному уроці повинна бути встановлена пріоритетність одного із предметів, тобто, бінарні уроки інформатика–математика і математика–інформатика можуть бути принципово різними. Компоненти уроку, який має вищий пріоритет називають *домінантними*, а іншого – *підпорядкованими*. Хоча бінарні уроки в класичному розумінні проводяться двома учителями, ми вважаємо, що сучасний бінарний, повинен проводити, як правило, один учитель.

Нижче представлено зразки бінарних уроків.

Урок 1

Тема : ПЕРЕТВОРЕННЯ ГРАФІКІВ ФУНКЦІЙ.

- Мета:** 1) Розглянути із учнями найважливіші засоби побудови графіків функцій за допомогою геометричних перетворень.
Сформулювати у учнів початкові вміння і навички у побудові графіків більш складних функцій.
- 2) Розвивати творчу ініціативу, логічне та абстрактне мислення учнів. вміння узагальнювати навчальний матеріал.
- 3) Виховувати самостійність у роботі, пізнавальний інтерес.

Тип уроку: засвоєння нових знань (бінарний урок інформатика-математика)

Обладнання: Комп'ютери, комп'ютерна програма "GRAND"
Підручник з алгебри, 7-9 клас, Г.П.Бевз, К., "Освіта", 1997

П Л А Н У Р О К У

I. ПІДГОТОВЧА ЧАСТИНА.

а) Активізація опорних знань учнів

На попередніх уроках було повторено основні відомості про функції і систематизовано їх у таку таблицю (у кожного з учнів є власна таблиця, див. таб.1).

Пригадати вивчені функції та їх графіки.

1. Перелічити відомі функції;
2. Описати графік прямої пропорційності;
3. Що необхідно відображати на схематичному графіку функції (точки перетину з осями, особливі точки)?
4. Побудувати схематично графік лінійної функції $y=3x-2$ (на дошці) та графіки функцій $y=-3x+2$ $y=x-1$ (на місцях, поваріантно, із взаємоперевіркою);
5. Описати та схематично побудувати на дошці графік оберненої пропорційності;
6. Порівняти графіки функцій $y=x^2$; $y=x^2$; $y=\sqrt{x}$. (*)

б) Мотивація.

Для чого вивчати графіки функцій? Де вони можуть знадобитись у майбутньому? Чи можна використати графіки функцій (*) для побудови графіків складніших функцій? На ці питання будуть дані відповіді на цьому уроці.

Тема уроку: Перетворення графіків функцій (демонструється з допомогою графопроектора для запису в зошитах).

Завдання уроку - виявити деякі закономірності графіків та навчитись їх застосовувати при побудові графіків складніших функцій з допомогою геометричних перетворень.

Для цього можна скористатись послугами інформатики, зокрема комп'ютерною програмою *Gran1*, яка допоможе будувати графіки функцій.

в) Повторення особливостей роботи з Gran1.

1) *Інтерфейс користувача* (головне меню, команди: ОБ'ЄКТ-СТВОРИТИ/ЗМІНИТИ, ГРАФІК – ПОБУДУВАТИ, ГРАФІК – МАСШТАБ – МАСШТАБ КОРИСТУВАЧА, панель інструментів, вікна: Графік, Список об'єктів, Відповіді, Введення функції, Властивості графічного вікна) – див. мал. 1-3:



мал. 1



мал. 2



мал. 3

2) Лінійний запис математичних виразів:

математична форма:	лінійна форма:
$1 + \frac{2}{3 + \frac{4}{5x}}$	$1 + 2 / (3 + 4 / (5 * x))$

3) 3
графік фун-

демонстра-
тора мал.



мал. 4

допомогою *Gran1* побудувати
кції $y = 1 + \frac{2}{3 + \frac{4}{5x}}$ (для звірки

ція з допомогою графопроек-
4):

Звернути увагу учнів, що з допомогою *Gran1* можна легко побудувати графік функції, яку навіть ще не вивчали (див. 2)).

II. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

За допомогою графопроєктора демонструються частини нового матеріалу, але без висновків, які учні записують їх у зошит, залишаючи після кожного пункта місце для висновків, щоб зробити їх у кінці уроку (див. додаток 1).

1. $y=f(x)$ та $y=-f(x)$
2. $y=f(x)$ та $y=f(x)+n$, $n>0$, $n<0$
3. $y=f(x)$ та $y=kf(x)$, $0<k<1$, $k>1$
4. $y=f(x)$ та $y=f(x-m)$, $m>0$, $m<0$

З допомогою графопроєктора демонструється завдання 1, такі ж завдання учні отримують на картках і виконують за комп'ютерами, працюючи в парах.

Завдання 1. Побудувати в одній системі координат та порівняти графіки функцій: $y=x^2$ та $y=-x^2$.

(мал.5, графіки симетричні відносно осі x)

Завдання 2. $y=x^2$ та $y=x^2+3$, $y=x^2-2$.

(мал. 6, графік піднявся на 3 одиниці вгору в напрямку осі y ; графік опустився на 2 одиниці вниз по осі y)

Завдання 3. $y=x^2$ та $y=\frac{x^2}{5}$, $y=3x^2$.

(мал. 7, значення функції зменшилося в 5 разів, графік стиснувся до осі x в 5 разів)

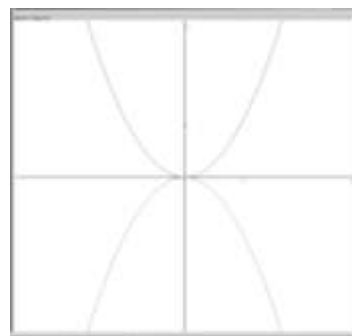
(значення функції збільшилося в 3 рази, графік розтягнувся від осі x в 3 рази)

Завдання 4. $y=x^2$ та $y=(x-3)^2$, $y=(x+2)^2$.

(мал. 8, графік змістився на 3 одиниці вправо в напрямку осі x)

(графік змістився на 2 одиниці вліво вздовж осі x).

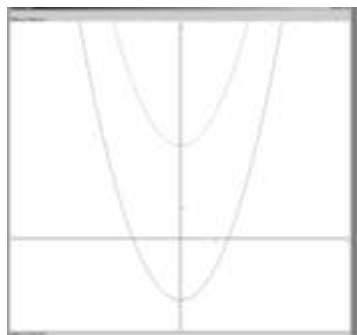
Учні виконують побудову, порівнюють графіки функцій і після обговорення записують у зошит висновки. Потім вони працюють за партами,



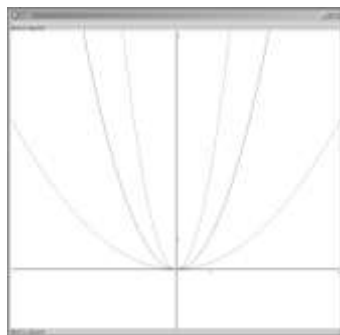
мал. 5

учитель демонструє додаток 1 і клас робить узагальнення для функцій $y=f(x)$ та записує висновки до кожного пункту. Учитель за допомогою графопроектора демонструє додаток 2, учні звіряють з ним свої записи в зошитах.

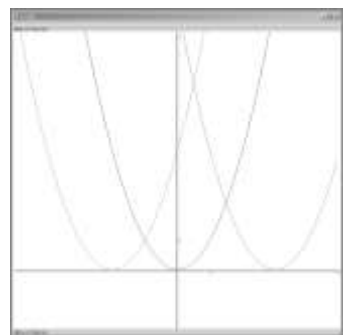
додаток 1.



мал. 6



мал. 7



мал. 8

додаток 2.

1. Графіки функцій $y=f(x)$ та $y=-f(x)$ симетричні відносно осі x .
2. Щоб отримати графік функції $y=f(x)+n$, потрібно графік функції $y=f(x)$ перенести на n одиниць в напрямку осі y , якщо $n>0$, або в протилежному напрямку, якщо $n<0$.
3. Щоб отримати графік функції $y=kf(x)$, потрібно графік функції $y=f(x)$ розтягнути від осі x в k разів, якщо $k>1$, або стиснути його в $1/k$ разів, якщо $0<k<1$.
4. Щоб отримати графік функції $y=f(x-m)$, потрібно графік функції $y=f(x)$ перенести на m одиниць в напрямку осі x , якщо $m>0$, або на $-m$ одиниць в протилежному напрямку, якщо $m<0$.

III. ЗАКРІПЛЕННЯ НОВОГО МАТЕРІАЛУ.

Розв'язати вправи:

№121. (усно) Як взаємно розміщені графіки функцій?

- a) $y=2x$ і $y=-2x$;
- б) $y=1/3x$ і $y=-1/3x$;
- в) $y=x^3$ і $y=-x^3$;
- г) $y=1/x$ і $y=-1/x$; (Графіки симетричні відносно осі x .)
- д) $y=x+2$ і $y=-(x+2)$;
- е) $y=\sqrt{x}$ і $y=-\sqrt{x}$;

№123 (усно) Як можна побудувати графік функції?

- a) $y=3x^2$; (розтягнути від осі x в 3 рази графік функції $y=x^2$)
- б) $y=1/2x^2$; (стиснути до осі x в 2 рази графік функції $y=x^2$)
- в) $y=0,5x^3$; (стиснути до осі x в 2 рази графік функції $y=x^3$)
- г) $y=-3x^2$; (симетрично графіку функції $y=3x^2$ відносно осі x)
- д) $y=-1/2x^2$; (симетрично графіку функції $y=1/2x^2$ відносно осі x)
- е) $y=-0,5x^3$; (симетрично графіку функції $y=0,5x^3$ відносно осі x)

IV. ПІДСУМКИ УРОКУ.

V. ДОМАШНЄ ЗАВДАННЯ.

Урок 2

Тема: РОЗВ'ЯЗУВАННЯ РІВНЯНЬ ГРАФІЧНИМ СПОСОБОМ

Мета: Розглянути методи побудови графіків функцій засобами діаграм Microsoft Excel, на основі цього навчити розв'язувати алгебраїчні рівняння графічним способом.

Тип уроку: засвоєння нових знань (бінарний урок математика-інформатика)

Обладнання та література:

- 1) М.І. Шкіль, З.І. Слєпкань, Алгебра і початки аналізу, посібний підручник для учнів 10-11 класів середньої школи, Київ, "Зодіак-еко", 2004 р.;
- 2) І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов, Інформатика, частина II, підручник для 10-11 класу загальноосвітніх навчальних закладів, Київ, "Форум", 2004 р.;
- 3) Microsoft Office Excel 2003.

ПЛАН УРОКУ

6. Повторення

1.1 Повторення знань з математики

а) Повторити визначення лінійного рівняння.

Для лінійного рівняння $a \cdot x = b$ можливі три випадки:

- 1) $a \neq 0$, тоді $a \cdot x = b \Leftrightarrow x = \frac{b}{a}$ єдиний корінь рівняння;
- 2) $a=0, b=0$, тоді рівняння $a \cdot x = b$ набуває вигляду $0 \cdot x = 0$, що правильно при будь-якому x , тобто коренем рівняння є будь-яке дійсне число;
- 3) $a=0, b \neq 0$, тоді рівняння $a \cdot x = b$ набуває вигляду $0 \cdot x = b$, воно не має коренів.

б) Повторити Функція $y=x^2$

Графіком функції є крива, яка має назву парабола. Складається з двох віток, симетричних відносно осі Оу. Найнижча точка параболи (початок координат) називається вершиною параболи.

Властивості функції $y=x^2$

1. Область визначення – множина $\mathbb{R}$ всіх дійсних чисел.
2. Область значень - множина невід'ємних дійсних чисел ($y \geq 0$).
3. Функція парна, тобто $y(-x) = y(x)$. Графік симетричний відносно осі Оу.

4. Якщо $x = 0$, то $y = 0$.
5. Монотонність: при $x > 0$ функція зростає, при $x < 0$ функція спадає.
6. Проміжки з-накосталості: при $x \neq 0$ у $y > 0$. Від'ємних значень функція не набуває.

в) Повторити властивості і графік функції **$y = \sin x$**

1. Область визначення – уся числова пряма, тобто **$D(\sin x) = \mathbb{R}$** .
2. Область значень – відрізок **$[-1; 1]$** , тобто **$E(\sin x) = [-1; 1]$** .
3. Функція **$y = \sin x$** - непарна, тобто **$\sin(-x) = -\sin x$** ; графік симетричний відносно початку координат.
4. Функція періодична з основним періодом **$T = 2\pi$** .
5. Нулі функції: **$\sin x = 0$** при **$x = k\pi$** , **$k \in \mathbb{Z}$**

Графік функції **$y = \sin x$** називається синусоїдою

г) Повторити властивості і графік функції **$y = \cos x$**

1. Область визначення – уся числова пряма, тобто **$D(\cos x) = \mathbb{R}$** .
2. Область значень – відрізок **$[-1; 1]$** , тобто **$E(\cos x) = [-1; 1]$** .
3. Функція **$y = \cos x$** – парна, оскільки **$\cos(-x) = \cos x$** ; графік симетричний відносно осі Оу.
4. Функція періодична з основним періодом **$T = 2\pi$** .

5. Нулі функції: **$\cos x = 0$** при **$x = \frac{\pi}{2} + k\pi$** , **$k \in \mathbb{Z}$**

Графік функції **$y = \cos x$** називається косинусоїдою.

1.2 Повторення знань з інформатики

д) Повторити роботу в програмі *Excel*: введення даних до комірки з допомогою прогресій, вставку формул у комірки, побудову діаграм і графіків.

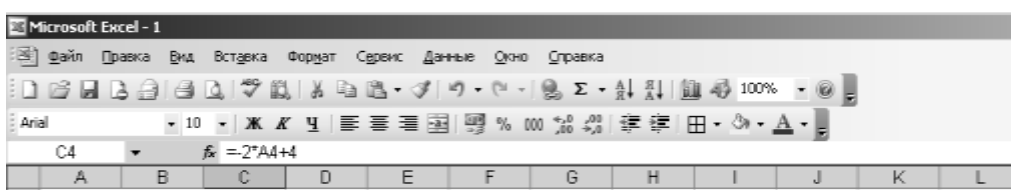
Звернути увагу на те, що суттєво спростити введення даних до електронної таблиці можна за допомогою засобу авто заповнення. Якщо потрібно ввести порядкові числа, дні тижня, місяці року або інші списки даних, то немає необхідності вводити кожний елемент окремо. Достатньо ввести кілька перших значень (задати крок), виділити діапазон комірок із значеннями, навести вказівник мишки на маркер авто заповнення – маленький чорний квадратик, розташований у правому нижньому куточку виділеного діапазону та виконати протягування мишкою на стільки клітинок, скільки їх треба заповнити даними.

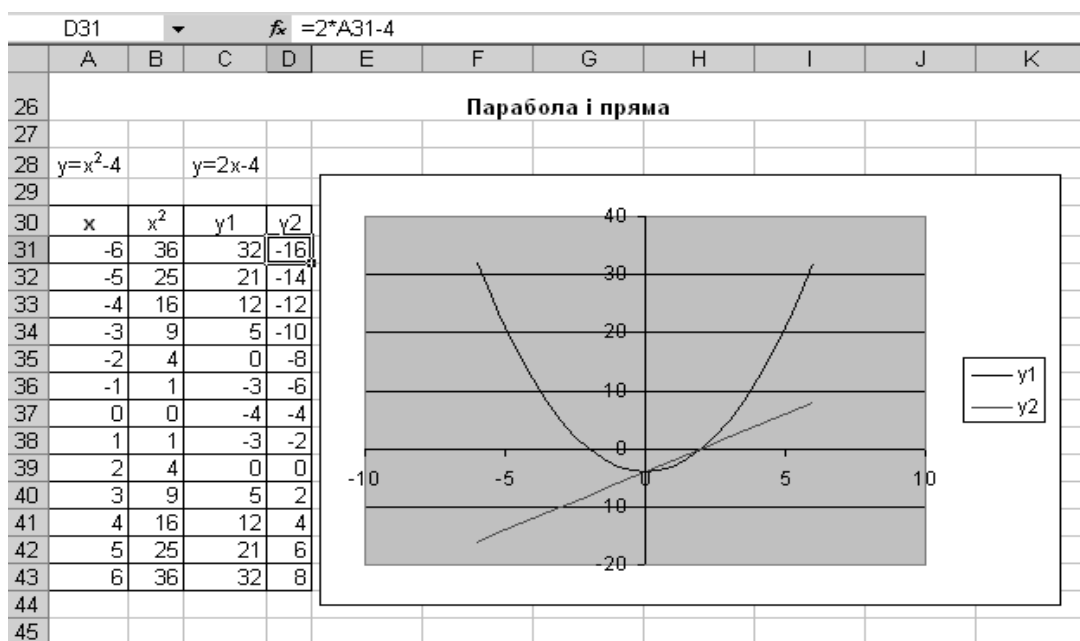
Також при побудові графіків слід звернути увагу на виділення потрібного діапазону комірок. Виділений діапазон не повинен містити порожніх комірок. Чому?

7. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ

2.1 Розглянути на прикладах розв'язування рівнянь **$3x - 6 = -2x + 4$** , **$x^2 - 4 = 2x - 4$** аналітичним способом

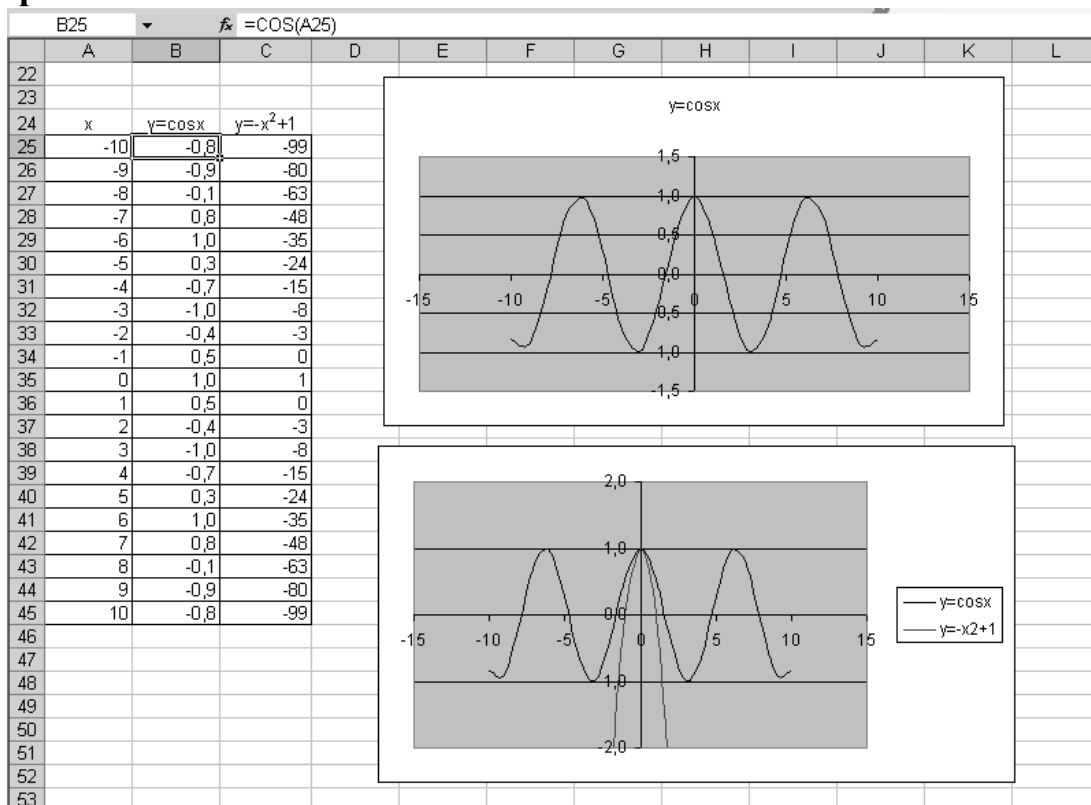
2.2 Розв'язування цих рівнянь графічним способом





2.3 Розв'язування графічним способом рівняння $\cos x + x^2 = 1$

2.4 Закріплення



Розв'язати рівняння *графічним способом*:

а) $x^2+x-6=0$

б) $x^2-4x+3=0$

в) $\frac{6}{x}=x-5$

г) $\frac{3}{x}-x+3=0$

д) $\sin x=x$

8. ДОМАШНЄ ЗАВДАННЯ

Розв'язати рівняння $x^2 - |x| = 0$ алгебраїчним і графічним способом.

Урок 3

Тема: ЗАСТОСУВАННЯ МАЙСТРА ПОБУДОВИ ДІАГРАМ MICROSOFT EXCEL ДЛЯ ПОБУДОВИ ГРАФІКІВ ФУНКЦІЙ.

Мета: Навчити учнів будувати графіки функцій, використовуючи майстер діаграм Microsoft Excel, дослідити перетворення графіків функцій, вміти форматувати діаграми.

Тип уроку: бінарний урок інформатика-математика

Обладнання:

- 1) І.Т. Зарецька, А.М. Гуржій, О.Ю. Соколов, Інформатика, частина II, підручник для 10-11 класу загальноосвітніх навчальних закладів, Київ, "Форум", 2004 р.;
- 2) Презентація "Створення діаграм";
- 3) Таблиця "Графіки функцій $y=\sin x$; $y=\sin x$; $y=\sin x$ "
- 4) Microsoft Office Excel 2003.

ПЛАН УРОКУ

1. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ

Фронтальне опитування:

1. Що таке функція?
 2. Які способи задання функції ви знаєте?
 3. Як побудувати графік функції?³⁶
4. Які типи даних використовують в ЕТ?
 5. Як ввести формулу?
 6. Що таке маркер заповнення?
 7. Що розуміють під поняттям форматування?

2. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

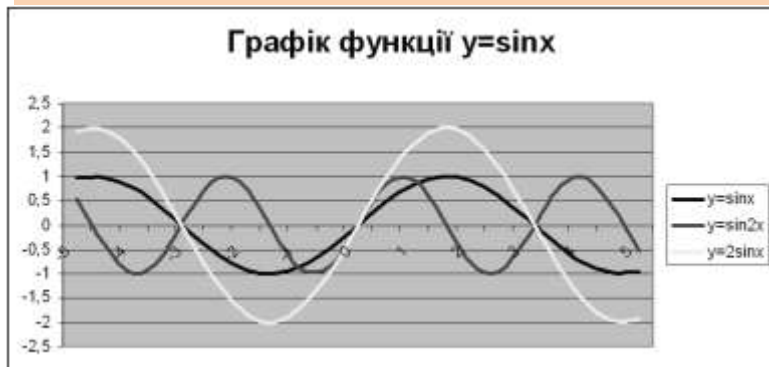
³⁶ В рамку будемо виділяти запитання або поняття з математики

Робота з презентацією "Створення діаграм" (додаток 1)

Проаналізувати графіки функцій та сформулювати алгоритм побудови графіків 2 та 3, використовуючи графік 1 :

1) $y = \sin x$, 2) $y = \sin 2x$, 3) $y = 2\sin x$.

3. ЗАСТОСУВАННЯ НОВИХ ЗНАНЬ ТА ВМІНЬ.



1. Побудувати графіки функцій: $y = \cos x$,

(1) $y = \cos \frac{x}{2}$,

(2) $y = \frac{1}{2} \cos x$.

(3)

2. Сформулювати алгоритм побудови графіків функцій 2 та 3, використовуючи графік 1.

3. Побудувати з допомогою Microsoft Excel графіки функцій (для зразка див. додаток 2)

$y = \sin x$, $y = |\sin x|$, $y = \sin|x|$.

4. ПІДВЕДЕННЯ ПІДСУМКІВ УРОКУ.

1. Що потрібно зробити, щоб побудувати діаграму чи графік?
2. Як користуватися майстром діаграм?
3. Як підписати графіки функцій в таблиці «Легенда»?
4. Як підписати значення осей координат даними з таблиці?
5. Що входить до поняття форматування діаграми?

6. Що розуміють в математиці під поняттям перетворення графіків функцій?

5. ДОМАШНЄ ЗАВДАННЯ.

Побудувати графіки функцій $y = \cos x$, $y = |\cos x|$ та записати алгоритм побудови графіка 2, використовуючи графік 1

Урок 1.

тема: ЗАГАЛЬНІ ВІДОМОСТІ ПРО КОМП'ЮТЕРНІ МЕРЕЖІ

мета:

РОЗГЛЯНУТИ:

- комп'ютерні мережі;
- засоби роботи в локальній мережі;
- глобальна комп'ютерна мережа Internet.

ВВЕСТИ ПОНЯТТЯ:

- модем;

- *ІІІ - адреса;*
- *сайт;*
- *Web- сервер;*
- *Web- сторінка;*
- *електронна пошта (e-mail);*
- *телеконференції;*
- *комп'ютерні бесіди.*

П Л А Н У Р О К У

I. ОРГАНІЗАЦІЙНИЙ МОМЕНТ.

II. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ.

III. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

1) ЗАГАЛЬНІ ВІДОМОСТІ ПРО КОМП'ЮТЕРНІ МЕРЕЖІ

Комп'ютерні мережі є ефективним засобом обміну інформації (програмами, даними) між різними комп'ютерами, об'єднання їх ресурсів та розділ навантаження. В залежності від відстаней між об'єднаними в мережу комп'ютерами вони поділяються на такі класи:

- локальні мережі, в яких комп'ютери знаходяться на невеликих відстанях, наприклад, в одному приміщенні або будівлі;
- регіональні мережі, що об'єднують комп'ютери в межах десятків і сотень кілометрів, наприклад, в містах, області;
- глобальні мережі, які забезпечують об'єднання комп'ютерів на великих відстанях, зокрема в різних країнах. Сучасною є тенденція входження локальних і регіональних мережах в глобальні.

Локальні мережі характеризуються високою швидкістю передачі даних та низьким рівнем помилок. Регіональні мережі до цих переваг додають ще й велику географічну протяжність. В глобальних мережах в даний час швидкість передачі даних і надійність нижча, але відсутні обмеження у відстані між комп'ютерами мережі.

З'єднання комп'ютерів вимагає відповідної апаратури, засобів зв'язку і програмного забезпечення для підключення до локальної мережі комп'ютера має бути оснащеним мереженою платою, а з'єднання комп'ютерів між собою здійснюється за допомогою кабелю або пари проводів. З'єднання комп'ютерів в глобальній мережі здійснюється по лініям телефонного або супутникового зв'язку з використанням **модема** – пристрою, який перетворює сигнали двійкових кодів у електричні сигнали низької частоти. Програмне забезпечення мережа включає драйвери мережних пристроїв та мережні додатки . для дотримання певного порядку обміну даними між комп'ютерами існують певні набору правил, які називають протоколом.

Кожен комп'ютер в мережі є її окремим вузлом . В мережі можуть бути виділені комп'ютери –сервери ,які виконують функції перевірки прав доступу до ресурсів мережі, управління обміном даних, зберігання спільних даних .

Інші комп'ютери мережі служать робочими станціями. Іменовану сукупність комп'ютерів в мережі називають доменом. Домен може бути поділеним на менші групи комп'ютерів які називають робочими групами.

2) ЗАСОБИ РОБОТИ В ЛОКАЛЬНІЙ МЕРЕЖІ

Операційна система **Windows** має вбудовані засоби роботи в локальних мережах Microsoft та Novell. Завдяки їм з будь-якого комп'ютера стає можливим доступ до папок чи інших ресурсів іншого комп'ютера мережі. Такі можливості надає папка **Сетевое окружение**, розміщена на **Робочому столі**. Відкрити цю папку можна також за допомогою утиліти **Проводник** або папки **Мой комп'ютер**.

Папка **Сетевое окружение** показує всі заявлені в робочій групі комп'ютери і значок для всієї мережі. Через неї можна встановити зв'язок з іншим комп'ютером мережі, в тому числі й іншої робочої групи. Кожному комп'ютеру в мережі відповідає певне ім'я, яке ідентифікує його в робочій групі. Імена та інші параметри мережі задаються під час інсталяції мережі за допомогою програми конфігурування, знаходиться в папці **Панель управління**, значок **Сеть**, закладка **Конфигурация**. Ці операції, як правило, виконуються досвідченими користувачами.

Для входу на деякий доступний комп'ютер мережі слід клацнути на його значку в папці **Сетевое окружение**. Це дає змогу побачити дозволені для спільного використання ресурси (диски, папки, принтери,). Робота за допомогою папки **Сетевое окружение** нагадує роботу з папкою **Мой комп'ютер**. Вона дозволяє переписувати файли з комп'ютера на комп'ютер, запускати програми на виконання тощо.

Використання ресурсів іншого комп'ютера можливе лише за наявності доступу до них. Доступ можна обмежувати, задаючи паролі або вказуючи категорії користувачів, яким доступ дозволений. Управління доступом до папки, файлів або пристроїв комп'ютера можна здійснити за допомогою утиліти **Проводник**. Для цього, завантаживши програму **Проводник**, необхідно викликати контекстне меню на папці, для якої має бути змінений спосіб доступу, вибрати команду **Доступ...**, У діалоговому вікні розкрити вкладенку **Доступ** і встановити в ній необхідні опції сумісного використання папки. Зокрема, опція **Локальний ресурс** забороняє сумісне використання папки.

3) ГЛОБАЛЬНА КОМП'ЮТЕРНА МЕРЕЖА INTERNET

Internet є найпопулярнішою всесвітньою комп'ютерною мережею, яка стала на даний час наймасовішим і оперативним джерелом інформації, розваг, найпрогресивнішим засобом спілкування і комунікації, сприятливим простором для бізнесу і реклами. Доступ власникам комп'ютерів до Internet надають фірми і організації, які називають провайдерами.

Кожен підключений до **Internet** комп'ютер має свою IP- адресу (IP- **Internet Protocol**), яка складається з розділених крапками чотирьох чисел із значеннями від 0 до 255. Для зручності **IP- адреси** кожного комп'ютера в **Internet** ставиться у відповідність унікальне ім'я, утворене з послідовності імен

доменів різного рівня, що ведуть до потрібного комп'ютера наприклад, **usttu@ck.ukrtel.net** - електронна адреса. Перший домен в цьому імені (usttu)- ім'я конкретного комп'ютера (абонента мережі), наступний (ck)- ім'я сервера (домена), до якого прикріплений даний комп'ютер, далі йдуть імена доменів вищих рівнів, закінчуючись доменом верхнього рівня (net), з якого починається пошук абонента. Домени верхнього рівня мають імена, які відповідають їх призначенню або країні (див. таблицю)

домен	використанням	домен	країна
Edu	Навчальні заклади	Ua	Україна
Com	Комерційні організації	Ru	Росія
Gov	Державні установи	Us	США
Net	Мережеві ресурси	Ca	Канада
Mil	Військові установи	Uk	Англія
Org	Інші організації	...	...

Комп'ютерна мережа **Internet** надає своїм абонентам наступні послуги:

- **доступ до інформаційних ресурсів мережі.** Найпопулярнішою системою, яка забезпечує такий доступ, на даний час є **WWW(World Wide Web-** Всесвітня павутина). В її основі лежить концепція **гіпертексту** – елемента документа (значка, малюнка, виділеного певним чином фрагмента тексту тощо), який служить посиланням на інші пов'язані з ним документи. Такий документ з елементами гіпертексту, розміщений на **Web- сервері**, називають **Web- сторінкою**. Групу Web- сторінок, зв'язаних між собою, називають **сайтом**. Адреса Web- сторінки складається з кількох елементів, наприклад:

[http://www. List.ru/catalog/10220.html](http://www.List.ru/catalog/10220.html)

де:

- **http://**- необов'язковий префікс, який показує комп'ютеру – маршрутизатору, що відбувається звернення до Web- сторінки, тобто гіпертекстового документа (http- Hyper Text Transfer Protokol, протокол передачі гіпертексту);
- **www.list.ru**- доменне ім'я сайту, під яким він зареєстрований в **Internet** (префікс **www** вказує на те, що мова йдеться про Web- сторінку);
- **catalog/10220/html** вказує на конкретну сторінку на потрібному сайті. Останні букви (**tml** або **htm**)- розширення файлу, яке вказує, що він належить до гіпертекстових документів.
- **електронна пошта (e-mail)**- система розсилки і отримання електронної кореспонденції. Кожин користувач **Internet** отримує у провайдера разом з правом доступу у мережу особисту електронну адресу (див. вище) і особисту «поштову скриньку» на сервері провайдера, на яку можуть приходити і зберігатися до вилучення користувачем від користувачів **Internet**. Існують також сервери, на яких можна відкрити безкоштовну поштову скриньку, зокрема **www.hotmail.com**, **www.yahoo.com**, **www.mail.ru**.

- телеконференції (**Usenet news** – система груп новин) мережений сервіс, орієнтований на підтримку колективних дискусій і подання оперативної інформації. Для спілкування використовуються дошки оголошення , де можна розмістити своє унікальне ім'я і являє собою об'єднання користувачів за інтересами.
- комп'ютерні бесіди (**IRC Internet Relay Chat**) – обмін повідомленнями стає одразу видимим всім учасникам бесіди.

IV. ПРАКТИЧНЕ ЗАВДАННЯ

Завдання. Створить на диску A: за допомогою програми Блокнот невеликий текстовий файл з іменем Лист від № XX, де XX- номер вашого столу в класі. Зробіть диск A: комп'ютера, на якому працюєте, доступним для інших комп'ютерів мережі. Ввійшовши за допомогою папки Сетевое окружение на сусідній комп'ютер, зробіть копію аналогічного файлу на свій диск A:. Відкрийте його і познайомтесь зі змістом.

V. Засвоєння вивченого теоретичного матеріалу

VI. Домашнє завдання.

VII. Підбиття підсумків уроку.

Урок 2.

тема: Програмне забезпечення роботи Internet

мета:

РОЗГЛЯНУТИ:

- **призначення, інтерфейс та налаштування програм-браузерів;**
- **формати Web- сторінок.**

Базові поняття і терміни: WWW, гіпертекст, браузер.

П Л А Н У Р О К У

I. ОРГАНІЗАЦІЙНИЙ МОМЕНТ.

II. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ.

III. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

Для роботи в **Internet** використовуються спеціальні програми, серед яких відмітимо:

- **браузери-** програми для перегляду Web- сторінок, наприклад, **MS Internet Explorer** та **Netscape Navigator** ;
- **Outlook Express-** програми для роботи з поштою і новинами;
- **NetMeeting-** програми для **Internet** – телефонії;

- **Chat**- програма для спілкування в текстовому режимі (IRC);
- **ActiveMovie**- програма для відтворення файлів мультимедіа – форматів ;
- **RealPlayer**- програма відтворення звукового потоку в форматі **RealPlayer (Internet radio)**;
- **NetShow**- програма для відтворення файлів формату Active Streaming Format (.ast) і реальних потоків ASF(**Internet**- телебачення);
- **FrontPage Express**- редактор Web- сторінок.

Нижче розглядається популярний браузер **Internet Explorer**. Його завантаження здійснюється або через команду меню **Пуск /Програми**, або через відповідний значок на **Робочому столі**. Вікно **Internet Explorer** містить стрічку меню, панель інструментів, адресна стрічка та вікно відображення сторінки.

В адресній стрічці вводяться адреси Web- сторінок. Якщо деяка сторінка відвідувалась, при вводиті її адреси **Internet Explorer** пропонує адресу по першим введенням буквам. Кнопка зображенням трикутника в кінці адресної стрічки розкриває список всіх відвіданих раніше сторінок.

Стрічка меню містить наступні пункти:

- **Файл** – призначений для роботи з файлами. Команда **Создать** дозволяє відкрити нове вікно **Internet Explorer**, сварити повідомлення електронної пошти послати листа в групу новин. Команда **Сохранить все** дає можливість записати вибрану сторінку на диск у вибраному форматі, зокрема в форматі **HTML**. Команда **Печать** здійснює друкування поточної сторінки принтером. Команда **Отправить** відправляє посилання на поточну сторінку по електронній пошті або в папку **Избранное**.
- **Правка** – призначена для редагування сторінки. Команда **Страница** відкриває сторінку в редакторі **FrontPage Express** для її правки. Команда **Найти на этой странице** дозволяє знайти на сторінці потрібне слово або вираз.
- **Вид** – служить для зміни виду **Internet Explorer**, зокрема вибору панелей інструментів (**Панель**), виводу списку раніше відведених сторінок (**Журнал**).
- **Переход**- призначений для переходів між Web- сторінками. Команда **Назад** повертає нас на попередню сторінку, а **Вперед** – на сторінку з якої відбулося повернення. Команда **Мой компьютер** відкриває вікно для роботи з локальними ресурсами комп'ютера, а **Адресная книга** відкриває адресну книгу для роботи з адресами.
- **Избранное** – містить посилання на відібрані користувачем сторінки **Internet**.

Панель інструментів містить кнопки, які служать для більш швидкого введення відповідних команд меню. Зокрема, **Остановить** зупиняє завантаження сторінки; **Обновить** повторно викликає сторінку; **Домой** переходить на основну сторінку; **Поиск** відкриває встановлену сторінку пошуку; **Каналы** відображає панель каналів; **На весь экран** розкриває вікно **Internet Explorer**

на весь екран; **Почта** відкриває підменю команд роботи з поштою і новинами; **Шрифти** змінює настройки шрифтів.

Якщо на сторінці зустрічаються гіперпосилання, то при наведенні на нього курсору миші останній приймає вигляд кисті руки з витягнутим пальцем, а в стрічці стану в нижній частині вікна **Internet Explorer** з'являється адреса відповідної Web- сторінки. Клацання лівою кнопкою мишки на гіперпосиланні приводить до переходу на сторінку, а правою – викликає контекстне меню з командами **Открыть**, **Открыть в новом окне**, **Сохранить объект как**, **Печать объекта**, **Копировать ярлык**, **Добавить в избранное**, **Свойства**.

IV. ПРАКТИЧНЕ ЗАВДАННЯ

Завдання. *Завантажте Internet Explorer та познайомтесь з його інтерфейсом. Змістом. Перейдіть на пошуковий сайт <http://www.rambler.ru> і перейдіть в ньому на сторінку, присвячену вашій майбутній професії, чи на будь-яку, яка вас більше цікавить.*

V. ЗАСВОЄННЯ ВИВЧЕНОГО ТЕОРЕТИЧНОГО МАТЕРІАЛУ

Гра «Круглий стіл»(Учні одержують запитання і по черзі дають відповіді)

Перелік можливих запитань:

1. Дайте класифікацію комп'ютерних мереж і їх коротку характеристику.
2. Яке обладнання необхідне для забезпечення роботи комп'ютера в локальних і глобальних мережах?
3. Що називають сервером? Доменом?
4. Яке призначення папки **Сетевое окружение**?
5. Як змінити доступ до комп'ютера для інших комп'ютерів локальної мережі?
6. Дайте опис електронних адресів користувачів **Internet**.
7. Які послуги **Internet** своїм абонентам?
8. Що розуміють під гіпертекстом і Web- сторінки?
9. З яких елементів складається адреса Web- сторінки?
10. Що являє собою електронна пошта?
11. Що розуміють під телеконференціями?
12. Які програми використовуються для роботи в **Internet**?
13. Якими способами в **Internet Explorer** можна викликати нову Web - сторінку?
14. Назвіть призначення кнопок панелі інструментів **Internet Explorer**.

VI. ДОМАШНЄ ЗАВДАННЯ.

VII. ПІДБИТТЯ ПІДСУМКІВ УРОКУ.

Урок 3.

Тема: ОСНОВИ РОБОТИ З ЕЛЕКТРОННОЮ ПОШТОЮ

Мета:

РОЗГЛЯНУТИ:

- *пошукові системи мережі Інтернет;*
- *настройка загальних параметрів Outlook;*
- *робота «Учетной записи»;*
- *робота з електронною поштою.*

П Л А Н У Р О К У

I. ОРГАНІЗАЦІЙНИЙ МОМЕНТ.

II. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ.

III. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

1. ЗАПУСК, НАСТРОЙКА, ІНТЕРФЕЙС. ОСНОВНІ ПАПКИ OUTLOOK

Запуск. Для того, щоб завантажити програму **Outlook Express** необхідно двічі клацнути на робочому столі на піктограму **Outlook**.

Інтерфейс. Оскільки **Outlook** містить в собі декілька різних елементів, потрібно вяснити, як відбувається переключення між ними і як змінюються функції меню, панелі інструментів та клавіатури в залежності від обраного в даний момент елементу програми.

Як в більшості програм, у верхній частині вікна **Outlook** розміщена стрічка меню, а під нею – панель інструментів. Додатково до цих стандартних елементів інтерфейса в **Outlook** існує ще одна панель, розташована в лівій частині вікна, яка називається панель **Outlook**.

Основні папки

Панель **Outlook** призначена для вибору вбудованих в **Outlook** елементів таких як папки: **Входящие, Календар, Контакти** та ін. Справа від панелі **Outlook** розміщене вікно перегляду інформації. Якщо в панелі **Outlook** вибрати папку **Входящие**, у вікні перегляду інформації буде список електронних повідомлень, які знаходяться в цій папці. Перемикання між елементами виконується з допомогою піктограм, які знаходяться на панелі **Outlook**.

Можна також переміщуватись між елементами з допомогою меню **Переход**.

При переміщенні між компонентами **Outlook** стрічка меню і панель інструментів також змінюється в залежності від обраного компонента. Крайня зліва кнопка панелі інструментів змінюється особливим чином, відображаючи вибір даного компонента. При виборі папки **Контакти** вона перетворюється в кнопку **Создать контакт**. Тобто, ця кнопка призначена для створення нового елемента даних компонента.

Настройка загальних параметрів Outlook

Ключовим моментом в отриманні максимальної віддачі від **Outlook** є його настройка, з метою забезпечення кращої роботи. Для установки параметрів, загальних для **Outlook**, існує діалогове вікно Параметри. Для того щоб відкрити це вікно необхідно вибрати команду Сервіс Параметри. Це діалогове вікно включає в себе велику кількість параметрів і складається з 11 вкладок. Крім цього, в ньому можна настроїти:

- Роботу **Outlook** с конфігураціями.
- Служби, проверяемые на наличие электронной почты.
- Уведомлени о прибытии новых сообщений.
- Параметры отправляемых сообщений.

Спосіб повідомлення про отримання нових повідомлень можна вибрати у вкладці **Электронная почта**. Можна ввімкнути повідомлення про отримання пошти, короткочасне змінення курсору миші, подачу звукового сигналу, чи будь-яку комбінацію з вказаного.

На вкладці Отправка можна вказати параметри шрифту та розміру символів, а також значення полів Важность та Пометка, використовувані по замовченню при створенні нових повідомлень.

Крім того, можна вказати, чи необхідно повідомляти про доставку прочитання повідомлень, Ці параметри працюють тільки в локальній мережі і можуть створювати велику кількість додаткових повідомлень в папці Входящие. Їх можна встановити також для окремих повідомлень, доставку і прочитання яких необхідно від слідкувати.

Для швидкої настройки і конфігурування **Outlook** можна використовувати праву кнопку миші, при натисканні якої практично на будь-якому об'єкті чи піктограмі виводиться контекстне меню. Якщо в ньому існує пункт Свойства, даних об'єкт можна налаштовувати.

2.РОБОТА З «УЧЕТНЫМИ ЗАПИСЯМИ».

Додавання «Учетной записи»для пошти або новин

Для додавання «Учетной записи» вам потрібно одержати у свого постачальника послуг Інтернету чи адміністратора локальної обчислювальної мережі наступні зведення:

- Для додавання поштової «Учетной записи» необхідно знати тип вашого поштового сервера (POP3, IMAP чи HTTP), ім'я і пароль для входу в систему, ім'я сервера для вхідної пошти і для протоколів POP3 і IMAP, ім'я сервера для вихідної пошти.
 - Для додавання «Учетной записи» новин необхідно знати ім'я сервера новин, з яким потрібно буде з'єднуватися, і, якщо потрібно, ім'я і пароль для входу на сервер.
1. Виберіть **Учетная записи** в меню **Сервис**.
 2. У діалоговому вікні **Учетные записи в Интернете** натисніть кнопку **Добавить**.

3. Виберіть **Почта** чи **Новости**, щоб почати роботу майстра підключення до Інтернету. Дотримуйтесь інструкції майстра, щоб налаштувати з'єднання із сервером пошти або новин.

Поради

- Кожен користувач може задати кілька облікових записів чи пошти новини. Для цього йому потрібно повторити вищевказані дії для кожного облікового запису.

Безкоштовно одержати поштовий обліковий запис можна в поштовий службу Hotmail, що використовує сервери HTTP.

IV. ПРАКТИЧНЕ ЗАВДАННЯ

Робота з програмою-браузером в онлайні. Знайти інформацію.

Текстову :

1. Прогноз погоди.
2. Курс валют (гривні, долара, євро).
3. Реферат з історії.

Графічну:

1. Карту Київської області.
2. Вітальну листівку.

Музичну:

1. Гімн України.
2. Улюблену мелодію.

Звіт оформити за допомогою текстового редактора (скріншоти сторінок із знайденою інформацією).

V. ЗАСВОЄННЯ ВИВЧЕНОГО ТЕОРЕТИЧНОГО МАТЕРІАЛУ

VI. ДОМАШНЄ ЗАВДАННЯ.

VII. ПІДБИТТЯ ПІДСУМКІВ УРОКУ.

Урок 4.

Тема: **РОБОТА З ЛИСТАМИ: СТВОРЕННЯ, ВІДПРАВЛЕННЯ ПОВІДОМЛЕНЬ, ПРИЄДНАННЯ ФАЙЛІВ ДО ЛИСТІВ.**

МЕТА:

РОЗГЛЯНУТИ:

- *робота з листами;*
- *відправлення листів, створення власної скриньки.*

ПРОДОВЖУВАТИ/ФОРМУВАТИ ВМІННЯ:

- *роботи з браузерами;*
- *пошуку інформації в мережі Інтернет;*
- *використовувати ресурси локальної та глобальної мережі;*
- *копіювати й переміщувати об'єкти в локальній мережі.*

ПЛАН УРОКУ

I. ОРГАНІЗАЦІЙНИЙ МОМЕНТ.

II. АКТУАЛІЗАЦІЯ ОПОРНИХ ЗНАНЬ.

III. ВИВЧЕННЯ НОВОГО МАТЕРІАЛУ.

1) ЕЛЕКТРОННА ПОШТА

За допомогою електронної пошти можна посилати і приймати повідомлення і файли по мережі **Internet** або локальній мережі. Повідомлення, які отримуються, вміщуються в папку Входящие. Вони представлені у вигляді таблиці, в якій вказуються автор повідомлення, його тема, дата і час отримання, а також інформація про те, чи є належні файли і чи було повідомлення прочитане.

Повідомлення відкриваються з допомогою подвійного натискання лівої клавіші мишки. При цьому в окремому вікні виводиться повний текст повідомлення. Після його прочитання можна продовжити роботу декількома способами: просто закрити вікно повідомлення, відповісти на повідомлення, переслати його кому-небудь ще, віддрукувати його і вилучити належні файли.

Створення та відправка електронного повідомлення

Для створення нового повідомлення потрібно з допомогою панелі **Outlook** відкрити папку Входящие і клацнути по клавіші панелі інструментів **Создать сообщение**. Після цього відкривається вікно нового повідомлення. Курсор буде встановлений в полі **Кому**, і можна або ввести електронну адресу з клавіатури, або вибрати її із тих, що зберігаються в адресній книзі, клікнувши на кнопці **Кому**.

Далі необхідно вказати тему повідомлення і написати його текст. Після того як ви набрали текст повідомлення необхідно клацнути по кнопці панелі інструментів **Отправить**. Якщо комп'ютер не під'єднаний до мережі, повідомлення буде переміщене в папку **Исходящие** і знаходиться там до тих пір, поки не буде проведено підключення.

Приєднання файлів до листів

Окрім звичайного тексту, до електронного повідомлення можна додавати файли різних типів, а потім відправляти та приймати їх разом з повідомленням. Повідомлення з вкладеними файлами відмічаються піктограмою канцелярської скріпки зліва від імені автора повідомлення.

Для того щоб приєднати файли до листа, клацніть на будь-якому місці у вікні створюваного повідомлення.

1. У меню **Вставка** виберіть **Вложение файла**.

2. Виберіть файл і натисніть кнопку **Вложить**.

Ім'я вложеного файлу з'явиться на полі **Вложить**, що знаходиться в заголовку повідомлення.

Примітка

- У повідомлення можна додавати текстові файли (файл з розширенням *.txt), вибравши в меню **Вставка** пункт **Текст из файла**.

Вставка малюнка у повідомлення

1. Виберіть місце в повідомленні, у якому потрібно вставити малюнок.
2. У меню **Вставка** виберіть **Рисунок** і натисніть кнопку **Обзор**, щоб вибрати файл малюнка.
3. Знайдете розташування і поля малюнка.

Примітка

- Якщо не вдається вибрати пункт **Рисунок**, перевірте, чи включене використання формату HTML. Для включення формату HTML у вікні повідомлення виберіть меню **Формат** і ввімкніть **Формату HTML**.

- Якщо одержувачем ваших повідомлень не вдається переглянути вставлені малюнки, виберіть **Параметры** в меню **Сервис**. У вкладці **Отправить** виберіть **Настройка HTML** і перевірте, чи включений прапорець **Отправить рисунки вместе с сообщениями**. Після цього повторіть відправлення повідомлення.

Для додавання фонового малюнка в повідомлення виберіть меню **Формат** у вікні повідомлення, укажіть на **Фоновый** і виберіть **Рисунок**. Натисніть кнопку **Обзор** для пошуку файлу малюнка.

2) РОБОТА З ЛИСТАМИ: ПАПКА «ВХОДЯЩИЕ», ПЕРЕГЛЯД, ЗНИЩЕННЯ, ВІДНОВЛЕННЯ ПОВІДОМЛЕНЬ, ВІД'ЄДНАННЯ ФАЙЛІВ ВІД ПОВІДОМЛЕНЬ.

Папка «Входящие» містить отримані листи. У верхній частині вікна «Входящие» знаходиться список листів. Для того, щоб проглянути зміст необхідного листа, потрібно двічі клацнути на ньому. Інформація з повідомлення міститиметься в нижній частині вікна

Видалення повідомлення з облікового запису ІМАР

1. Виберіть повідомлення в списку повідомлень.
2. Натисніть кнопку **Удалить** на панелі інструментів. Повідомлення буде позначено як вилучене.
3. У меню **Правка** виберіть **Уничтожить удаленные сообщения**, щоб остаточно забрати вилучені повідомлення з папки. Знищене повідомлення не може бути відновлено.

Відновлення вилучених повідомлень

- Щоб відновити повідомлення, позначене як вилучене на сервері ІМАР, у меню **Правка** виберіть **Восстановить**.

- Щоб вилучені повідомлення автоматично знищувалися, у меню **Сервис** виберіть **Параметры**. У вкладці **Обслуживание** виберіть **Уничтожить удаленные сообщения при выходе из папок IMAP**.
- Для знищення повідомлень необхідно, щоб ви знаходилися в мережі. Якщо ви працюєте в автономному режимі, повідомлення будуть знищені при підключенні до мережі.

Видалення поштового повідомлення

1. У списку повідомлень виберіть необхідне повідомлення.
2. Натисніть кнопку **Удалить** на панелі інструментів.

Відновлення вилучених повідомлень

- Для відновлення вилученого повідомлення відкрийте папку **Удаленные** і перетягніть потрібні повідомлення в папку **Входящие** чи іншу.
- Якщо ви не хочете, щоб по закінченні роботи з програмою **Outlook Express** вилучені збереглися в папці **Удаленные**, виберіть **Параметры** в меню **Сервис**. У вкладці **Обслуживание** ввімкніть прапорець **Очищать папку (Удаленные)** перед виходом.

Щоб остаточно видалити усі вилучені повідомлення, відкрийте папку **Удаленные**. Виберіть **Очищать папку (Удаленные)** у меню **Правка**.

3) ВІДПРАВЛЕННЯ ПОВІДОМЛЕНЬ ЧЕРЕЗ INTERNET, СТВОРЕННЯ ВЛАСНОЇ ПОШТОВОЇ СКРИНЬКИ.

Пошту **Internet** можна використовувати для обміну повідомленнями з іншими людьми, які можуть знаходитися як в іншій частині світу, так і в сусідній кімнаті, незалежно від того чи приєднані вони до вашої мережі. Стандарти пошти **Internet** дають можливість комп'ютерам і мережам різних типів зв'язуватися між собою.

Для того щоб скористатися можливістю спілкування за допомогою пошти **Internet**, необхідно включати конфігурацію і настроїти службу пошти **Internet**, для цього потрібна інформація про поштовий сервер, який використовується в мережі або у провайдера послуг **Internet**. Мінімум, що необхідно знати, - це свою електронну адресу ім'я або IP – адрес поштового серверу, ідентифікатор користувача та пароль.

Для того щоб додати служби пошти **Internet**, необхідно:

1. Запустити **Outlook**.
2. За допомогою команди **Сервис + Службы** відкрити діалогове вікно **Службы**.
3. Клацнути по кнопці **Добавить** щоб відкрити діалогове вікно **Добавление служб** конфігурації і додати служби пошти **Internet**.
4. Вибрати в списку **Почта Internet** і клацнути на кнопку **ОК**, після цього буде виведено вікно **Почта Internet**.

5. У верхній частині вікна ввести ім'я та електронну адресу, а нижче – ім'я або ІП – адресу поштового сервера, ідентифікатор користувача і пароль. Дві кнопки, які розташовані внизу, використовуються для установки додаткових параметрів.

Створення власної поштової скриньки.

Для створення власної поштової скриньки необхідно звернутися до місцевого провайдера **Internet**, який надасть вам електронну адресу. Ця електронна адреса буде використовуватися на поштових серверах як назва вашої поштової скриньки.

Адреса користувача складається з двох частин: його ідентифікатора і назви домену, які розділюються @. У свою чергу, назва домену може складатися з декількох частин, відокремлених крапкою. Ці частини утворюють ієрархічну структуру, починаючи з найнижчого рівня. Наприклад, IP – адреса

Авраменко @ kneu. kiev.ua

Розшифруємо так :

Авраменко – назва комп'ютера користувача;

kneu – назва організації;

kiev – назва міста;

ua – скорочена назва країни.

IV. ПРАКТИЧНА РОБОТА.

1. Відкрити у верхній частині вікна «Входящие» де знаходиться список листів, відкрити будь-який лист видалити його і відновити.
2. Знайти поштову адресу школи і скопіювати її.
3. Створити лист приєднати до нього файл і відправити по локальній мережі.

V. ЗАСВОЄННЯ ВИВЧЕНОГО ТЕОРЕТИЧНОГО МАТЕРІАЛУ

Подумай і відповідай:

1. Як завантажити програму **Outlook Express**.
2. Як можна дізнатися про отримані повідомлення?
3. Що потрібно для додавання «Учетной записи»?
4. Скільки облікових записів може задати будь-який користувач?
5. Як створити і відправити електронне повідомлення?
6. Як приєднати файли до листів?
7. Як вставити малюнок у повідомлення?
8. Як видаляються поштові повідомлення?
9. Як відновлюються видалені поштові повідомлення?
10. Як створити власну поштову скриньку?

VI. ДОМАШНЄ ЗАВДАННЯ.

VII. ПІДБИТТЯ ПІДСУМКІВ УРОКУ

ВИКЛАДАННЯ ІНФОРМАТИКИ В 5-6 КЛАСАХ.**ПРО ПЛАНУВАННЯ КУРСУ ІНФОРМАТИКИ В 5-6 КЛАСАХ
ЗА НОВИМИ ПРОГРАМАМИ**

Тут мова буде про викладання інформатики згідно навчальної програми для 5-9 класів загальноосвітніх навчальних закладів з поглибленим вивченням предметів природничо-математичного циклу [1], хоча в заголовку вказано: "відповідно нових програм". Справа в тому, що ця програма була затверджена вслід за програмою [2], призначеною для загальноосвітніх навчальних закладів. По-перше, обидві програми реалізують ті самі ідеї нової концепції загальної середньої освіти та відповідного Державного стандарту базової і повної загальної середньої освіти. По-друге, вони обидві передбачають перехід до вивчення базового курсу інформатики в 5-9 класах. По-третє, у програмах [1] та [2] передбачена однакова кількість навчальних годин із деякими відмінностями у розподілі навчального часу між окремими темами та класами. По-четверте, за мету програми [1] у межах 5-6 класу не може ставитись поглиблене вивчення інформатики, адже враховуючи вік учнів та те, що програма передбачена для вивчення цього предмету в класах, де інформатика не вивчалась раніше, тут можливий лише пропедевтичний або вступний рівень розгляду навчального матеріалу.

Порівняльна характеристика програм [1] та [2] в обсязі 5-6 класів свідчить, що між ними якщо й існують відмінності у складності навчального матеріалу, то дуже незначні, адже курси 5-6 класів є фактично, як уже зазначено вище, пропедевтичними. Суттєва різниця між згаданими програмами полягає тільки в підходах до вивчення предмету інформатики, у [1] – чітко виражений змістовно-компетентнісний, тоді, як у [2] – фактично чисто компетентнісний. Обидва підходи до вивчення навчальних предметів сьогодні діють, очевидно, деякий час вони конкуруватимуть. Тут не ставиться за мету ні порівняння згаданих програм, ні їх протиставлення, але у контексті обговорення нової на даний час ідеї перенесення базового курсу інформатики з 9-11 класів у 5-9 класи, цілком природно і правомірно розглянути шанси змістовно-компетентнісного підходу.

Безсумнівно, коли існує дозвіл на використання програми [1], то слід розробити й видати відповідні підручник та методичні рекомендації щодо його застосування, адже без цього програма не може бути впроваджена належним чином. За мету цих рядків ставиться ініціювання процесу масового впровадження названої програми.

Слід ще раз зауважити, що обидві програми, розраховані на учнів, які не вивчали інформатики до 5 класу, а в умовах, коли з 2013-2014 навчального року паралельно вводиться курс інформатики в навчальні плани для 5-х і 2-х класів це означає, що обидві програми недовговічні, через 3 роки їх очевидно буде необхідно докорінно змінювати.

У зв'язку з цим тут розглянемо лише перший крок впровадження, навівши зразки календарно-тематичних планів для 5 та 6 класу згідно програми [1]. Для п'ятого класу таке планування потрібне вже сьогодні, щодо шостого, то на черзі стоїть завдання термінового написання підручника, а значить, також необхідне й відповідне календарно-тематичне планування.

Складання календарно-тематичних планів – це компетенція виключно вчителів, які будуть викладати навчальний предмет відповідно до даної програми. Але, як завжди, при переході до нових програм, а тим більше, до нових освітніх стандартів, багатьом учителям необхідна певна методична допомога. Така допомога стає ще необхіднішою, коли вчителі не мають відповідного підручника (відомо, що він у процесі підготовки до друку і вже отримав гриф "Рекомендовано"). Крім того, останній розділ програми "Виконання індивідуального проекту" не розшифрований погодинно, хоча на нього відводиться 5 навчальних годин, про цей розділ також немає інформації і в Пояснювальній записці. Вчителю досить важко спланувати такий розділ, якщо він включений у програму 5 класу, де учні мають ще дуже обмежений обсяг знань, умінь та навичок, необхідних для планування та

реалізації навчального проекту. Потребує також деякої конкретизації і розділ "Алгоритмізація й програмування".

У календарно-тематичному плані, на відміну від програми, необхідно розписувати кожен тему поурочно, а також, по можливості, скорочувати формулювання тем, не міняючи їх змісту, адже тему кожного уроку доведеться записувати учням у зошити, а вчителів у класному журналі, де для цього відводиться лише частина одного рядка. Виходячи із конкретних місцевих умов, календаризування програми передбачає деякі перестановки тем та у межах дозволеного перерозподіл годин між розділами, а при потребі, збільшення кількості практичних робіт. Саме такі особливості мають наведені нижче календарно-тематичні плани з інформатики відповідно до програми [1] (додатки 1 та 2).

З метою досягнення логічної цілісності курсу та зміцнення міжтематичних зв'язків в індивідуальному проекті (5 клас, див. додаток 1) доцільно застосувати практичні навички з трьох попередніх розділів, що забезпечить наступність і підсилить радіально-концентричну лінію, декларовану в Пояснювальній записці програми. Це пропонується реалізувати шляхом вибору конкретної теми проекту, пов'язаного з математичною постановкою задачі, розглядом блок-схеми алгоритму та застосуванням для побудови блок-схем алгоритмів програми Dia v0.97, що вільно копіюється з Інтернету. Додаткове введення опису алгоритму блок-схемою, цілком доступне п'ятикласникам, тому природно після словесного опису алгоритму познайомити учнів із блок-схемами. У програмі передбачено знайомство із середовищем програмування, у якості якого тут вибрано ALGO (автор В. Петрів). Але такий вибір зовсім не обов'язковий, адже зараз доступні й інші реалізації програмування на ознайомлювальному етапі, наприклад, середовище СКРЕТЧ. Проте, хоч воно й пропонується як візуальне об'єктно-орієнтоване середовище програмування для навчання школярів молодших і середніх класів, все ж має більш ігровий, ніж навчальний характер, цікавіше саме молодшим школярам, тому у межах базового курсу інформатики не може бути серйозним конкурентом чисто навчального середовища програмування, такого як, наприклад, АЛГО.

Після таких досить косметичних змін дійсно можна попрацювати над проектом, п'ятикласники будуть уже до такої роботи готові. Оформлення індивідуального проекту пропонується зробити у вигляді текстового документа, що додатково закріпить навички, набуті при вивченні розділу "Опрацювання тексту". Під час виконання проекту з метою полегшення роботи п'ятикласників над ним доцільно два уроки побудувати у формі практичних робіт № 11-12 (не передбачених програмою). З метою отримання можливості детальніше виконати індивідуальний проект цей розділ збільшено на 1 годину за рахунок резерву навчального часу. Три години також з резерву часу відведено для заключного повторення.

У календарно-тематичному плані для 5 класу дещо перероблений розділ "Алгоритмізація й програмування", чіткіше виражена змістовна лінія *АЛГОРИТМ – ПРОГРАМА - НАВЧАЛЬНЕ СЕРЕДОВИЩЕ ПРОГРАМУВАННЯ* з одного боку і відокремлено лінійні та циклічні алгоритми. Введено знайомство з виконавцем алгоритмів РОБОТ.

Подібні особливості має й зразок календарно-тематичного плану з інформатики для 6 класу (див. додаток 2). У зразку календарно-тематичного плану для 6 класу також додано за рахунок резерву по одній годині до розділів "Алгоритмізація й програмування" та "Виконання індивідуального проекту". У першому з названих розділів введено виконавець *ЧЕРЕПАШКА*.

В програмі [1] в змістовній лінії "Алгоритмізація й програмування" в 5 класі пропонується знайомство з циклічними алгоритмами, а в 6 класі із обчислювальними алгоритмами та алгоритмами із логічними виразами, в той час, як поняття величини програмою пропонується ввести в 7 класі. У зв'язку з цим для конкретизації пунктів програми в календарно-тематичному плануванні і було вибрано виконавці РОБОТ та ЧЕРЕПАШКА у відповідних місцях. Крім того, в 6 класі введено поняття величини, оголошення та опис величин. Система команд виконавця РОБОТ дозволяє фактично використовувати логічні вирази, не вводячи цих понять (вирази зразка ВільноПопереду). При використанні вико-

навця ЧЕРЕПАШКА в 6 класі введення величин та логічних виразів стає цілком зрозумілим і своєчасним після пропедевтичного використання логічних виразів (тверджень) у 5 класі.

Решту відмінностей з відповідними розділами програми пропонується шляхом співставлення виявити самостійно.

ДОДАТОК 1

КАЛЕНДАРНО-ТЕМАТИЧНИЙ ПЛАН З ІНФОРМАТИКИ ДЛЯ 5-А КЛАСУ

		I семестр
№	Назва теми	Примітки
Інформація й інформаційні процеси (2 години)		
1	Поняття інформації. Види інформації. Інформатика як наука.	
2	Інформаційні технології, їх застосування у повсякденному житті. Об'єкти та їх властивості.	
Основи роботи на комп'ютері (5 годин)		
3	Комп'ютер – пристрій для опрацювання даних і управління процесами. Складові комп'ютера. Правила безпечної роботи на персональному комп'ютері.	
4	Поняття програми та програмного забезпечення. Вмикання й коректне завершення роботи комп'ютера.	
5	Практична робота № 1. Умикання й коректне завершення роботи комп'ютера. Робота з клавіатурним тренажером.	
6	Робочий стіл та його елементи, панелі завдань, меню. Поняття файлу і папки. Запуск і завершення програм. Призначення основних клавіш. Клавіатурні тренажери.	
7	Практична робота № 2. Робота з вікнами та їх об'єктами.	
Опрацювання тексту (6 годин)		
8	Середовище текстового редактора. Введення тексту. Вставлення й змінювання символів. Поняття абзацу й фрагменту тексту.	
9	Практична робота № 3. Уведення й редагування тексту.	
10	Властивості тексту і його частин як об'єктів. Копіювання, перенесення й вилучення фрагментів. Форматування тексту.	
11	Практична робота № 4. Уведення й форматування тексту.	
12	Відкриття і збереження документа. Перевірка правопису.	
13	Практична робота № 5. Редагування й форматування тексту за зразком.	
Опрацювання зображень (4 години)		
14	Поняття растрової комп'ютерної графіки.	
15	Середовище графічного редактора. Панель інструментів. Графічні примітиви. Створення й редагування малюнка. Властивості малюнка як об'єкта.	
16	Практична робота № 6. Створення найпростіших зображень.	
17	Практична робота № 7. Створення й редагування зображень за зразком.	
		II семестр
Алгоритмізація й програмування (7 годин)		
18	Поняття алгоритму і його виконавців. Словесний спосіб описання алгоритму. Найпростіші лінійні алгоритми, їх розроблення й описання.	
19	Алгоритми з повторенням. Розроблення й описання алгоритму з повторенням.	
20	Навчальне середовище програмування. Призначення й основні можливості, які воно забезпечує. Виконавець Робот.	
21	Практична робота № 8. Ознайомлення з навчальним середовищем програмування.	

22	Практична робота № 9. Створення найпростіших лінійних програм у навчальному середовищі програмування для виконавця Робот.	
23	Практична робота № 10. Створення найпростіших циклічних програм у навчальному середовищі програмування для виконавця Робот.	
24	Залік по темі "Алгоритмізація й програмування".	
Виконання індивідуального проекту (6 годин)		
25	Математична постановка задачі та її опис у текстовому редакторі.	
26	Складання словесного алгоритму поставленої задачі.	
27	Блок-схема алгоритму. Середовище для побудови блок-схеми алгоритму. Dia v0.97.	
28	Практична робота № 11. Побудова блок-схеми, відповідної словесному алгоритму.	
29	Практична робота № 12. Оформлення індивідуального проекту у вигляді текстового документа.	
30	Залік по темі "Виконання індивідуального проекту".	
Повторення (3 години)		
31	Інформаційні процеси. Основи роботи на комп'ютері.	
32	Створення текстового документу із графічними зображеннями.	
33	Підсумковий урок.	
Резерв часу (2 години)		

ДО-
ДА-
ТОК
2
КА-
ЛЕ-
НДА
РНО
-
ТЕ-
МА
ТИ-
ЧНИ
Й
ПЛА
Н З
ІН-
ФО-
РМА
ТИ-

КИ ДЛЯ 6 КЛАСУ

		I семестр
№	Назва теми	Примітки
	Операційна система (3 години)	
1	Інтерфейс операційної системи, його об'єкти та робота з ними. Вікна, їх типи, призначення та елементи. Ярлики та дії з ними. Файли, їх імена, розширення, шаблони та типи. Шлях до файлу та повне ім'я файлу. Пошук файлів і папок. Копіювання, переміщення, вилучення та відновлення папок, файлів та їх груп.	
2	Практична робота № 1. Виконання операцій над папками і файлами.	
3	Практична робота № 2. Пошук і відновлення файлів та папок.	
	Опрацювання тексту (5 годин)	
4	Призначення та класифікація текстових процесорів.	
5	Створення, редагування, форматування і зберігання документів. Основні вимоги до оформлення тексту. Практична робота № 3.	
6	Пошук і заміна в тексті. Використання шрифтів, накреслень і кольорів у тексті.	
7	Вставка у текстові документи символів, графічних об'єктів та таблиць. Практична робота № 4.	
8	Перегляд текстового документу та підготовка його до друку. Практична робота № 5.	
	Опрацювання зображень (5 годин)	
9	Растрова і векторна графіка. Векторні зображення та принципи їх побудови.	
10	Побудова та властивості векторних зображень. Робота з контурами. Зафарбування векторних зображень.	
11	Практична робота № 6. Побудова графічних об'єктів засобами векторної графіки.	
12	Переміщення, обертання, копіювання та групування векторних зображень.	
13	Практична робота № 7. Додавання тексту до зображення.	

	Комп'ютерні презентації (5 годин)	
14	Комп'ютерні презентації та засоби їх створення і демонстрації.	
15	Створення презентації, вставка графічних і текстових об'єктів.	
16	Створення презентацій з допомогою майстра та шаблонів. Практична робота № 8.	
17	Гіперпосилання та кнопки управління та переходів. Практична робота № 9.	
18	Налаштування показу слайдів. Практична робота № 10.	
<i>II семестр</i>		
	Основи Інтернету (4 години)	
19	Загальні відомості про Інтернет. Поняття про браузер. Перегляд веб-сторінок. Система імен в мережі. Веб-сайти та веб-сторінки.	
20	Інтернет-енциклопедії. Поняття про хмарні сервіси. Пошук відомостей у мережах.	
21	Збереження отриманих даних. Інтернет-загрози. Поняття про достовірну, недостовірну та шкідливу інформацію. Захист даних та основні принципи безпеки.	
22	Практична робота № 11. Пошук відомостей в мережах Інтернет.	
	Алгоритмізація і програмування (6 годин)	
23	Об'єкти і способи керування ними у навчальному середовищі програмування АЛГО. Виконавець (об'єкт) Черепашка та система його команд. Програма як спосіб опису послідовності дій об'єкта.	
24	Програмування дій об'єкта, що повторюються. Практична робота № 12. Малювання замкненої лінії з допомогою Черепашки.	
25	Поняття величин, їх оголошення та опис. Обчислювальні алгоритми.	
26	Практична робота № 13. Розробка і опис обчислювальних алгоритмів.	
27	Алгоритм (програма) побудови зображень у навчальному середовищі програмування. Логічні операції.	
28	Практична робота № 14. Програмування руху об'єктів з використанням логічних операцій.	
	Виконання індивідуального проекту (4 години)	
29	Створення сценарію кількаслайдової презентації із меню, гіперпосиланнями та кнопками переходів, що демонструє рух Черепашки.	
30	Практична робота № 15. Створення слайда меню з гіперпосиланнями.	
31	Створення презентації згідно сценарію. Збереження презентації в режимі демонстрації.	
32	Залік по темі "Виконання індивідуального проекту".	
	Резерв (3 години)	

ЛІТЕРАТУРА:

1. "Навчальна програма. Інформатика. 5-9 класи загальноосвітніх навчальних закладів з поглибленим вивченням предметів природничо-математичного циклу" (За редакцією академіків НАПН України А.М. Гуржія і В.Ю. Бикова. Авторський колектив: Громко Г.Ю., Іванов Є.О., Лапінський В.В., Литвинова С.Г., Мельник В.І., Пасіхов Ю.Я., Руденко В.Д. Комп'ютер у школі та сім'ї, №6, 2012 р.)
2. "Інформатика. Навчальна програма для учнів 5–9 класів загальноосвітніх навчальних закладів" (автори Жалдак М.І., Морзе Н.В., Ломаковська Г.В., Проценко Г.О., Ривкінд Й.Я., Шакотько В.В.)

Тема: Поняття алгоритму та його виконавців

Мета: Дати поняття алгоритму, виконавця алгоритму та системи команд виконавця на прикладі виконавця Черепашка

ХІД УРОКУ

ЩО ТАКЕ АЛГОРИТМ?

Сьогодні, навіть не помічаючи цього, люди повсякчас користуються дуже важливим поняттям **"алгоритм"**. Це одне із найважливіших наукових понять.



Алгоритм – це сукупність дій, вказівок, інструкцій, чітке виконання яких у визначеному порядку приводить до виконання поставленого завдання.

ІСТОРИЧНА ДОВІДКА

Слово «алгоритм» походить від «algorithmi» – латинської форми написання імені великого узбецького математика **Мухаммеда аль-Хорезмі**, який приблизно 825 до н. е. написав трактат (наукову працю), в якому описав при-



думану в Індії позиційну десяткову систему числення і правила виконання арифметичних дій над ними. Тому спочатку під алгоритмом розуміли тільки правила виконання чотирьох арифметичних дій над багатоцифровими числами в десятковій системі числення. Зараз він є одним із фундаментальних понять інформатики. В першій половині XII століття книжка потрапила до Європи в перекладі латинською мовою під назвою *Algorithmi de numero Indorum*. Вважається, що перше слово в перекладі відповідає невдалій латинізації імені Аль-Хорезмі, а назва перекладу звучить як «Алгоритмі про індійську лічбу».

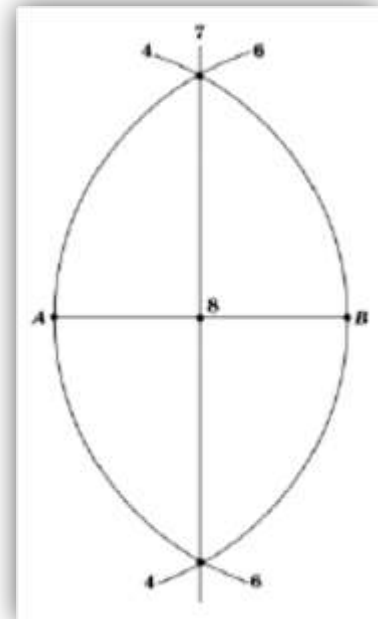
СЛОВЕСНА ФОРМА ПОДАННЯ АЛГОРИТМІВ

Алгоритми можуть бути описані в різних формах, а саме:

- у словесній формі;
- у вигляді блок-схеми;
- у вигляді програми.

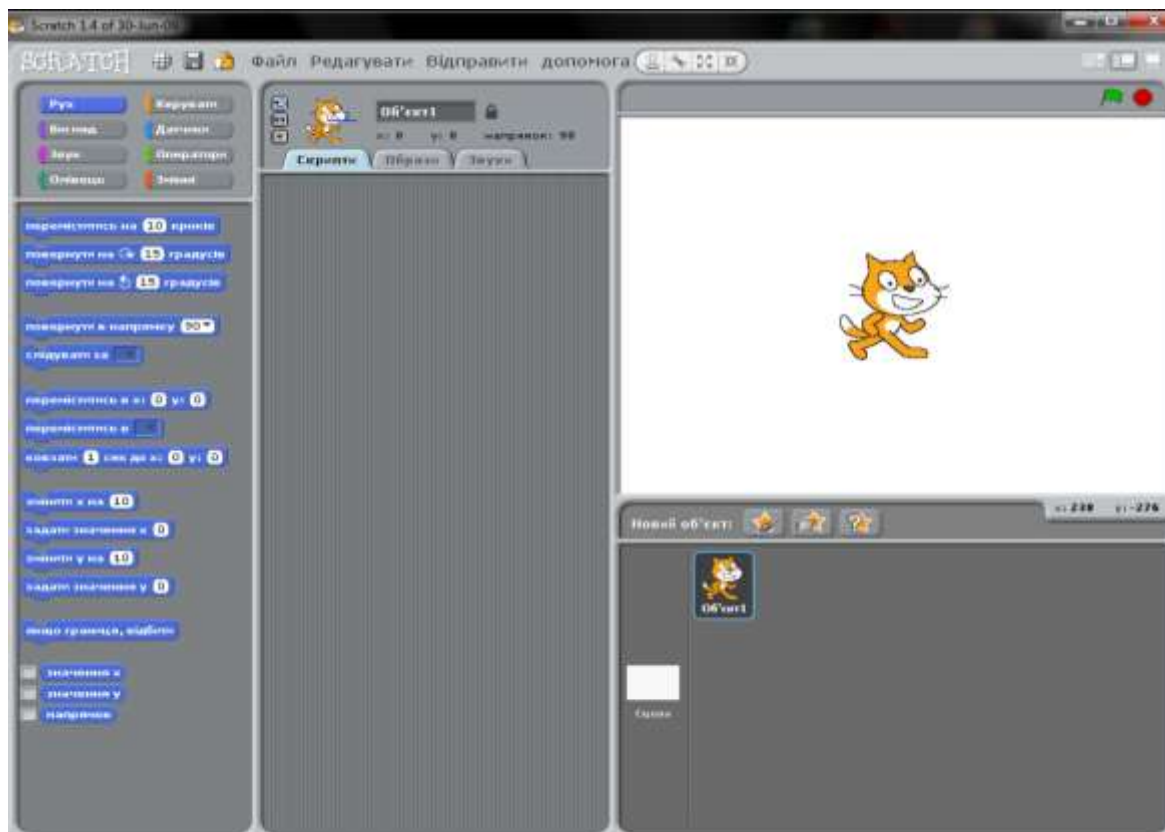
Ось приклад алгоритму поділу відрізка навпіл з допомогою циркуля та лінійки, записаний словесно:

1. Почати. П.2
2. Встановити ніжку циркуля в один кінець відрізка. П.3
3. Розкрити циркуль на довжину відрізка. П.4
4. Провести коло. П.5
5. Переставити ніжку циркуля у другий кінець відрізка. П.6
6. Провести коло. П.7
7. Провести пряму через точки перетину кіл. П.8
8. Позначити точку перетину проведеної прямої і відрізка. П.9
9. Кінець.

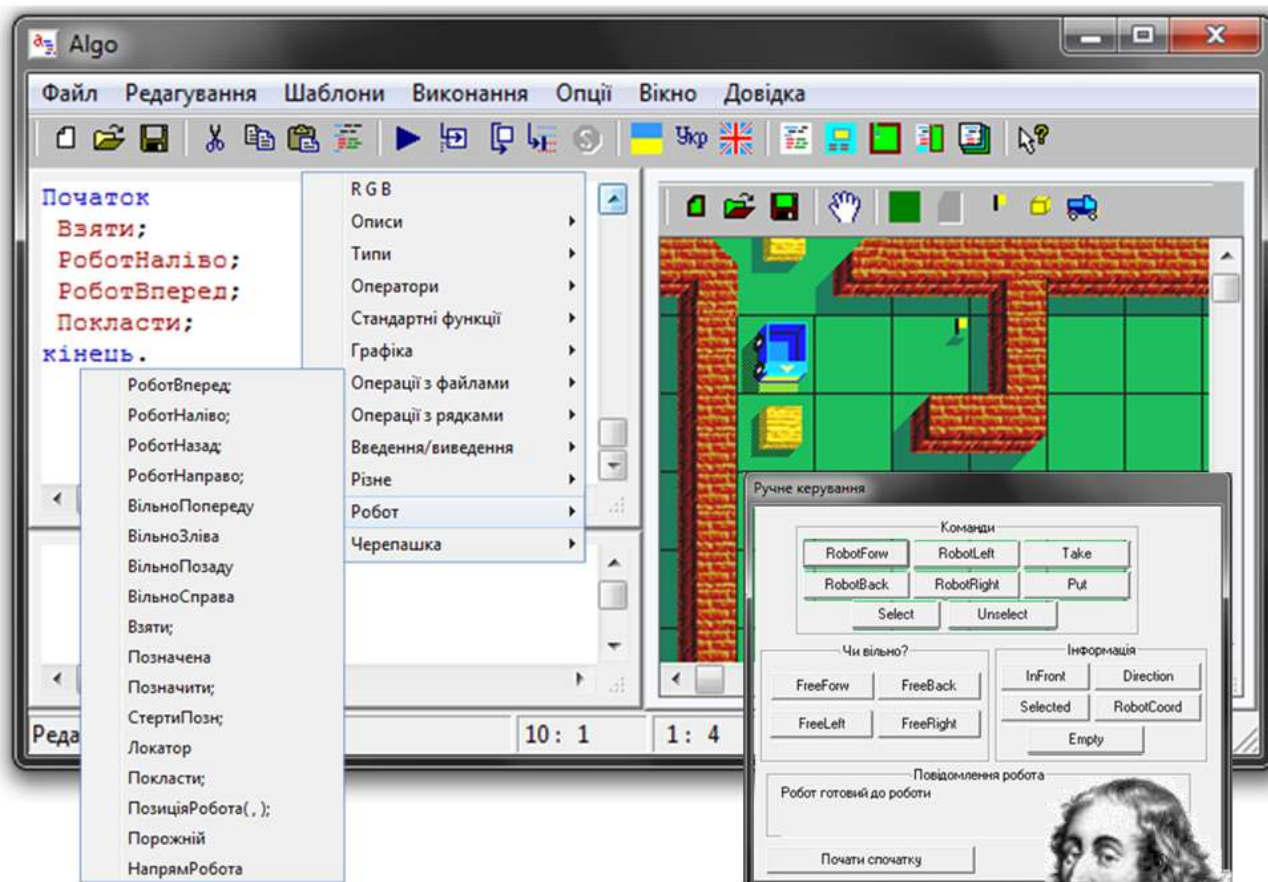


ВИКОНАВЦІ АЛГОРИТМІВ

Кожен алгоритм, як послідовність вказівок, призначений певному виконавцеві для реалізації. Наведений алгоритм поділу відрізка навпіл з допомогою циркуля та лінійки призначений для виконавця-людини, якого назвемо **КРЕСЛЯР**. Але виконавцями більшості алгоритмів є **комп'ютери**. Такі алгоритми називаються **програмами**. Існують комп'ютні програми, призначені для створювання і виконання алгоритмів. Такі програми називаються **системами** (середовищами) **програмування**. В Інтернеті можна відшукати ігрові, анімаційні середовища програмування, наприклад **SCRATCH** (СКРЕТЧ).

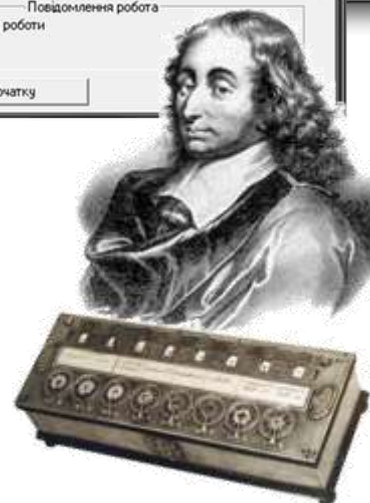


Ми познайомимося з більш придатним для вивчення у школі україномовним середовищем програмування АЛГО (версія 2.0.2, автор В. Петрів), яке вільно розповсюджується в Інтернеті.



В його основу покладена мова програмування Pascal³⁷, яка знадобиться нам у майбутньому. Крім того, інструментальне середовище АЛГО простіше в користуванні та краще пристосоване для знайомства з алгоритмами й програмами. АЛГО оснащено двома виконавцями: **РОБОТ** і **ЧЕРЕПАШКА**.

Кожен виконавець має свою *систему команд* (СКВ). На малюнку відображена СКВ РОБОТА. Для її демонстрації в АЛГО, як відображено на малюнку, вмонтований ручний режим управління виконавцем РОБОТ.



ПРИКЛАДИ ЛІНІЙНИХ АЛГОРИТМІВ.



Лінійним називають такий алгоритм, у якому всі дії (вказівки, інструкції) виконуються послідовно одна за одною від першої до останньої у порядку їх розміщення.

Наведений вище алгоритм поділу відрізка навпіл є лінійним.

³⁷ Названо в честь знаменитого французького математика й фізик Блеза Паскаля, який створив перший механічний пристрій для автоматизації розрахунків.

Нескладні алгоритми:

- ☐ додавання двох чисел;
- ☐ перехід регульованого перехрестя;
- ☐ визначення у класі самого високого учня;
- ☐ запалювання вдома газового пальника.

Складні алгоритми:

*плавлення металу в доменній печі;
стикування космічних кораблів;
хірургічна операція на серці;
складання розкладу руху потягів на залізниці.*

ОСНОВНІ ВЛАСТИВОСТІ АЛГОРИТМІВ

! Зауважимо, що не кожену послідовність інструкцій або команд можна назвати алгоритмом.

Алгоритм повинен відповідати певним вимогам, тобто мати відповідні властивості, основними з яких є такі:

- 1) **ДИСКРЕТНІСТЬ**³⁸ – полягає в тому, що алгоритм складається із скінченної кількості *окремих* команд;
- 2) **ЗРОЗУМІЛІСТЬ** – полягає в тому, що кожна команда алгоритма повинна входити у СК вибраного виконавця;
- 3) **ТОЧНІСТЬ** – полягає в тому, що серед команд алгоритму повинна бути встановлена черговість, тобто переходи;
- 4) **МАСОВІСТЬ** – полягає в тому, що виконання алгоритма забезпечує розв’язання будь-якої задачі даного типу;
- 5) **РЕЗУЛЬТАТИВНІСТЬ** – полягає в тому, що через *скінченне* число кроків (виконаних команд) та за *обмежений час* обов’язково отримується результат.

ЗАКРІПЛЕННЯ

Запитання:

- 1) *Сформулюйте алгоритм ділення відрізка прямої на дві рівні частини.*
- 2) *Які є способи описання алгоритмів?*
- 3) *Які алгоритми називають лінійними?*
- 4) *Наведіть приклади нескладних алгоритмів.*

Завдання:

- 1) *Опишіть, що таке алгоритм.*
- 2) *Опишіть алгоритм вимикання комп’ютера.*
- 3) *Опишіть алгоритм переходу регульованого перехрестя.*

³⁸ Дискретний – не суцільний, той, що складається з окремих частин.

- 4) *Опишіть алгоритм Вашого руху з дому до школи.*
- 5) *Опишіть алгоритм поділу площі круга за допомогою циркуля й лінійки на чотири рівні частини.*
- 6) *Запишіть алгоритм додавання двох десяткових чисел.*
- 7) *За допомогою циркуля й лінійки виконайте розміщення в колі радіусом R шести рівносторонніх трикутників так, щоб їх вершини знаходилися у центрі кола, а вони не перетиналися між собою. Опишіть алгоритм, заяким виконано побудову.*

Практична робота:

У середовищі АЛГО створити алгоритм (програму), який переносить вантаж згідно малюнку у позначене прапорцем місце.

ДОМАШНЄ ЗАВДАННЯ

П. 5.1, Завдання:

- 1) *Опишіть алгоритм побудови за допомогою циркуля й лінійки рівностороннього трикутника, сторона якого дорівнює заданому відрізьку.*
- 2) *На одній прямій стоять три футболісти. Опишіть алгоритм розміщення їх у порядку збільшення їхнього зросту.*

Запишіть алгоритм Вашого шляху до школи.

Тема: Виконавець Черепашка. Алгоритми з повторенням. Розроблення й описання алгоритму з повторенням.

Мета: Дати поняття алгоритму, виконавця алгоритму та системи команд виконавця на прикладі виконавця Черепашка.

ХІД УРОКУ

ПОВТОРЕННЯ

• Запитання:

- 1) Які команди повинні входити в СКВ КРЕСЛЯР?
- 2) Які властивості алгоритмів передбачають необхідність існування системи команд виконавця?
- 3) Які програми називаються лінійними?

• Практичне заняття із середовищем розробки алгоритмів АЛГО

- 1) Записати в зошиті алгоритм (програму) для виконавця РОБОТ, який виконує роботу, передбачену малюнком:
- 2) Виконати в ручному режимі дії, передбачені програмою, записаною в завданні 1).

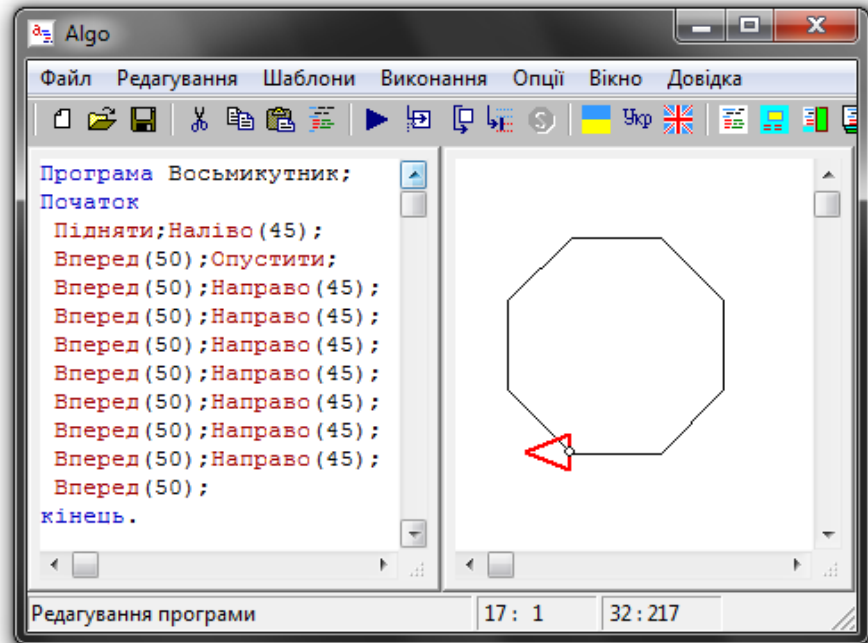


3) Написати та виконати в середовищі АЛГО програму, написану в завданні 1).

ВИКОНАВЕЦЬ "ЧЕРЕПАШКА"

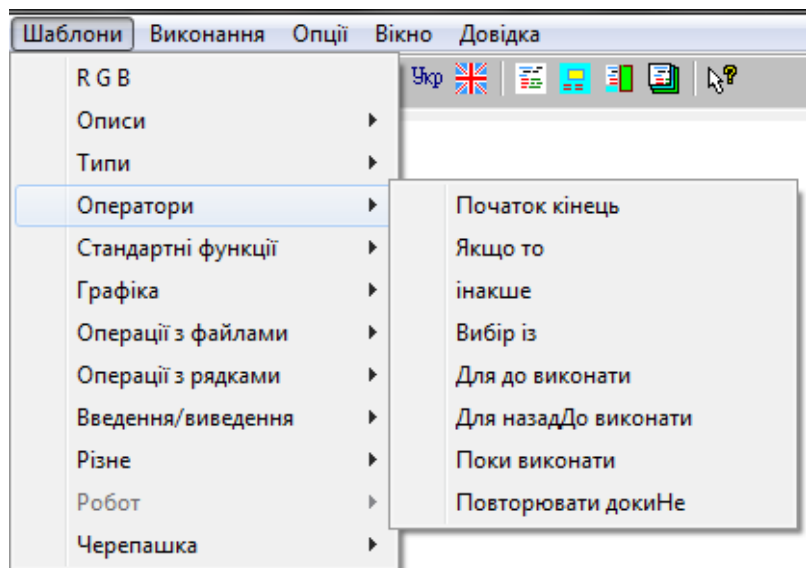
Черепашка – це популярний виконавець алгоритмів, який може з допомогою інструменту **ПЕРО**, відоме вам ще з роботи в графічному редакторі **Paint**, креслити потрібні лінії або створювати певні малюнки.

Як бачимо на малюнку, **ЧЕРЕПАШКА** в АЛГО зображена у вигляді червоного трикутника, схожого на стрілку. Вказівки програми легко прослідкувати за малюнком. Спробуйте виконати їх у зошиті, взявши одну клітинку за 50 кроків у вказівці руху, а у вказівці повороту маючи на увазі градусне вимірювання кутів.



АЛГОРИТМИ З ПОВТОРЕННЯМ

У наведеній програмі для **ЧЕРЕПАШКИ** по дві команди записано в одному рядку. Це зроблено для економії місця. Також видно, що ця програма лінійна. Але програма містить аж сім однакових рядків: **"Вперед (50); Направо (45);"**, тобто певні дії багаторазово повторюються. Для компактішого запису програм у **програмуванні**³⁹ передбачена спеціальна вказівка **повторення**. Вказівок (команд або операторів) повторення існує декілька видів:



³⁹ Програмування – це процес складання програм, він вивчається в спеціальній галузі інформатики з такою ж назвою.

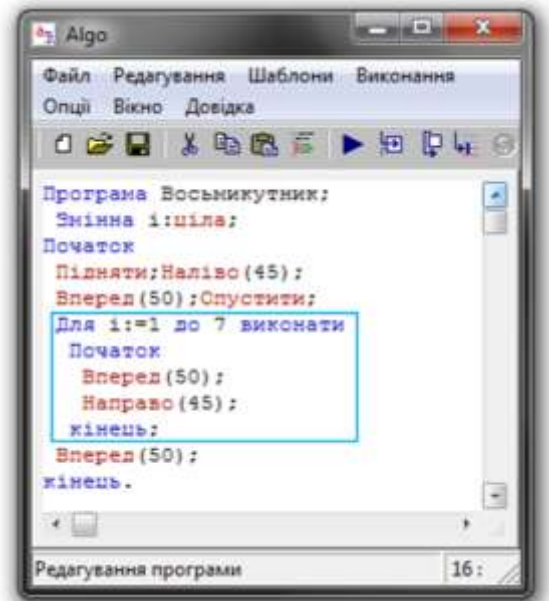
з *передумовою* (Поки виконати); з *післяумовою* (Повторювати доки не); з *параметром* (Для до виконати, Для назад до виконати).

Використавши вказівку повторення з параметром у програмі "Восьмикутник" (обведена синьою рамкою), вдається значно спростити й скоротити її запис (порівняйте із попереднім варіантом програми "Восьмикутник"). Зміст цієї команди такий: 7 разів повторити дві команди "*Вперед (50); Направо (45);*".

У цій команді є параметр, який визначає та контролює кількість повторень. У програму "Восьмикутник" параметром служить змінна величина *i*. Величини у програмуванні мають дещо інший зміст, ніж у математиці⁴⁰, перед використанням

у програмі їх слід оголосити й описати у спеціально відведеному місці – *розділі оголошень та описів*, яке знаходиться нижче *заголовку програми*, але перед записом вказівок, розміщених між словами "*Початок*" та "*кінець*."

У програмуванні кожна вказівка (команда, оператор) має *формат* – загальний запис та правило виконання.



! Формат команди повторення з параметром

ДЛЯ *параметр* := початкове значення **ДО** кінцеве значення **ВИКОНАТИ**

[**ПОЧАТОК**]

оператор (тіло цикла, серія команд)

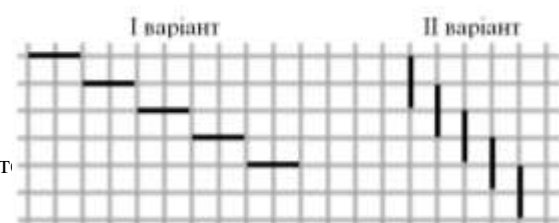
[**КІНЕЦЬ**];

Слова **ДЛЯ**, **ДО**, **ВИКОНАТИ**, **ПОЧАТОК** та **КІНЕЦЬ** називаються *службовими* (зарезервованими), їх запис і місцезорозміщення не можна змінювати. Записи у форматі команди, взяті в квадратні дужки можуть бути не завжди, а в окремих випадках (тоді, коли *оператор* складається біль, ніж із однієї команди). Назвіть зарезервовані слова, параметр, його початкове та кінцеве значення і оператор у команді повторення з параметром, розміщеній у програмі "Восьмикутник".

! Правило виконання команди повторення з параметром

Команда повторення з параметром виконується так:

- 1) Значенню параметра надається початкове значення;
- 2) Перевіряється *умова*: значення параметра не перевищує кінцевого значення.
- 3) Якщо *умова істинна*, то



⁴⁰ Поняття величини, постійної та змінної величини вивчались в мат

- виконується тіло цикла;
- значення параметра збільшується на 1;
- знову перевіряється умова.

Якщо *умова не істинна*, команда повторення з параметром вважається виконаною.



Умова – це твердження, яке може бути істинним або хибним.

ЗАКРІПЛЕННЯ

Ознайомлення із СКВ ЧЕРЕПАШКА

- Відкрити вікно середовища програмування АЛГО;
- Вивести на екран випадające меню із СКВ ЧЕРЕПАШКА;
- Переписати СКВ ЧЕРЕПАШКА в зошит;
- Порівняти СКВ ЧЕРЕПАШКА із СКВ РОБОТ.

Практична робота

- 1) Записати в зошиті циклічний алгоритм мовою PascalABC, який виконує малюнок;
- 2) У середовищі АЛГО створити циклічну програму, яка відображає малюнок.

ЗАПИТАННЯ ТА ЗАВДАННЯ

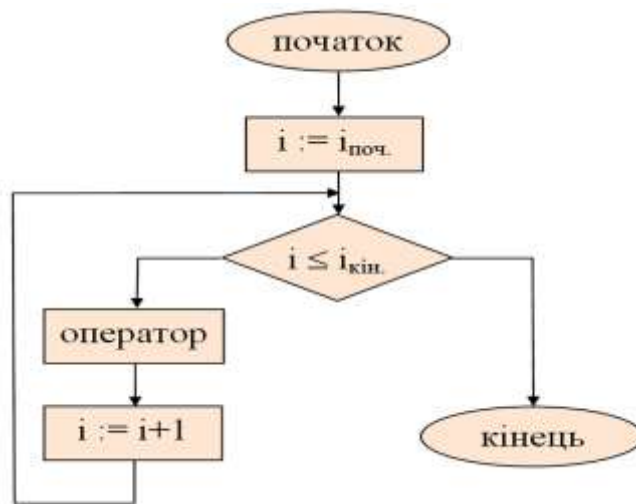
- 1) Який вигляд має виконавець ЧЕРЕПАШКА?
- 2) Який вигляд має робоча область (сцена) виконавця ЧЕРЕПАШКА?
- 3) Назвати прості вказівки виконавця ЧЕРЕПАШКА;
- 4) Назвати вказівки виконавця ЧЕРЕПАШКА без параметрів;
- 5) Які параметри можуть бути у вказівках виконавця?
- 6) Чому в СКВ ЧЕРЕПАШКИ окремі вказівки позначені синіми, а решта червоними позначками?
- 7) Назвати приклади тверджень;
- 8) Назвати приклади істинних тверджень;
- 9) Назвати приклади хибних тверджень;
- 10) Чи можуть бути твердження, істинність яких невідома?
- 11) Чи можуть бути твердження, які можуть бути істинними чи хибними залежно від обставин?

✓	Вперед();
✓	Назад();
✓	Направо();
✓	Наліво();
✓	Підняти;
✓	Опустити;
✓	Показати;
✓	Сховати;
✓	Додому;
✓	Напрям
✓	Скерувати();
✓	Позиція(,);
✓	НаПозицію(,);
✓	Азимут(,)



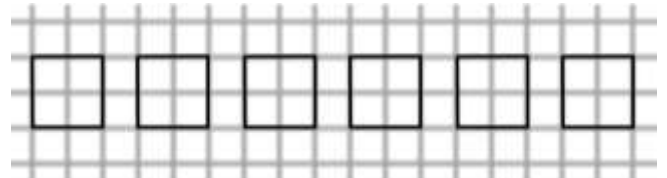
БЛОК-СХЕМА ВКАЗІВКИ ПОВТОРЕННЯ З ПАРАМЕТРОМ

Вказівку повторення з параметром для більшої наочності можна зобразити графічно, у вигляді блок-схеми.



ДОМАШНЄ ЗАВДАННЯ

- 1) Вивчити формат, блок-схему та правило виконання команди повторення з параметром;
- 2) Дати відповіді на питання 1-11;
- 3) Створити програму, яка зображує малюнок:



ЛІТЕРАТУРА:

1. Інформатика, навчальні програми для 10-11 класів загальноосвітніх навчальних закладів. «Прем'єр», Запоріжжя – 2003 р.
2. Інформатика, навчальна програма для учнів 9 класу загальноосвітніх навчальних закладів : Інформаційний збірник МОН України – № 19-21, липень 2009 р.
3. Остапець В. С. Погляд на майбутнє шкільної інформатики – «Комп'ютер у школі та сім'ї» – № 7, 2006 р.
4. Остапець В. С. Комплексні електронні дидактичні засоби в системі інформаційних технологій навчання : методичний посібник для вчителів – «Ри-зографіка» – Бориспіль – 2007 р.
5. Остапець В. С. Комп'ютерне моделювання й основи алгоритмізації та програмування у формі експрес-курсу : методичний посібник – «Інформатика» – № 10 – 2009 р.

6. Остапець В. С. Експрес-курс з алгоритмізації та програмування : навчальний посібник – «Інформатика» – №11-13 – 2009 р.
7. Остапець В. С. Письмовий опис алгоритмів як невід’ємна складова розв’язку задач з програмування в школі : стаття – «Комп’ютер у школі та сім’ї» – № 7 – 2003 р.
8. Остапець В. С. Шкільні олімпіади з інформатики : стаття – «Світло» – № 1– 2001 р.
9. Остапець В. С. Використання багаторозрядної арифметики : стаття – «Інформатика» – № 17 – 2005 р.
10. Остапець В. С. Від задачі до задачі : стаття – «Інформатика» – № 33-34 – 2005 р.
11. Остапець В. С. Метод динамічного програмування у школі : навчально-методичний посібник – «Інформатика» – №9 – 2008 р.
12. Остапець В. С. Позакласна робота з програмування у школі : навчально-методичний посібник – «Ризографіка» – Бориспіль – 2007 р.
13. Дьяченко В. К. Сотрудничество в обучении : О коллективном способе учебной работы. – М.: Просвещение – 1991 г.
14. Пометун О., Пироженко Л. Інтерактивні технології навчання : теорія і практика. – К. – 2002 р.
15. Зарецька І. Т., Гуржій А. М., Соколов О. Ю. Інформатика, частина 2 : підручник для 10-11 класів загальноосвітніх навчальних закладів – Київ – «Форум» – 2004 р.
16. Непомняща Т. Бінарний урок : стаття – «Інформатика» – № 10 – квітень 2001 р.
17. Жалдак М. І., Морзе Н. В., Мостіпан О. І., Науменко Г. Г. Інформатика та інформаційні технології в навчальних закладах, № 2-3, 2008 р. : програми для загальноосвітніх класів – Інформатика – 7-9 клас.
18. Ривкінд Й. Я., Лисенко Т. І., Чернікова Л. А., Шакотько В. В. Інформатика 9 клас : підручник для загальноосвітніх навчальних закладів – Київ, Генеза – 2009 р.
19. Морзе Н. В., Вембер В. П., Кузьмінська О. Г. Інформатика : підручник для 9 класу – Київ – Школяр – 2009 р.
20. Володіна І. Л., Володін В. В. Інформатика 9 : підручник для 9 класу загальноосвітніх навчальних закладів – Харків – Гімназія – 2009 р.
21. Хижа О. Л., Розв’язування задач підвищеної складності з інформатики : стаття – Київ – Інформатика – №№ 37-43 – 1999 р.
22. Рудик О. Б., Програма спеціального курсу «Прикладна математика» – КМІУВ ім. Б. Грінченка – 2000 р.
23. Савченко В. С., Разработка алгоритмов : от простого к сложному – Донецк – 1996 р.
24. Грузман М. З. Эвристика в информатике – Винница – Арбат – 1998 р.

25. Бондаренко В., Жук С. Задачі XII Всеукраїнської олімпіади з інформатики та обчислювальної техніки : стаття – Комп'ютер у школі та сім'ї – № 3 – 1999 р.
26. Остапець В. С. Сучасний бінарний урок як найвищий ступінь міжпредметних зв'язків : стаття – «Інформатика» – № 43 – 2009 р.
27. Остапець В. С. Про планування курсу інформатики в 5-6 класах за новими програмами : стаття – «Комп'ютер у школі та сім'ї» – № 7 – 2013 р.

ⁱ На основі матеріалу "Якою бути шкільній інформатиці завтра?", Комп'ютер у школі та сім'ї, №2, 2006р.

ⁱⁱ За методичним посібником «Комплексні електронні дидактичні засоби в системі інформаційних технологій навчання», м. Бориспіль, за сприяння Бориспільської районної Ради та благодійного фонду «Борисове поле», 2007р.