

Сумський державний педагогічний університет

Фізико-математичний факультет

Кафедра інформатики

Лі Чен

**ДИФЕРЕНЦІЙОВАНИЙ ПІДХІД В НАВЧАННІ УЧНІВ
ПРОГРАМУВАТИ НА УРОКАХ ІНФОРМАТИКИ У 10-11 КЛАСАХ**

Спеціальність: 014 Середня освіта (Інформатика)

Галузь знань: 01 Освіта/Педагогіка

Кваліфікаційна робота на здобуття освітнього ступеня магістра

Науковий керівник:

_____ В.Г. Шамоня,
кандидат фіз.-мат. наук, доцент,
доцент кафедри інформатики

«_____» _____ 2025 року

Виконавець:

_____ Лі Чен

«_____» _____ 2025 року

Суми 2025

ЗМІСТ

СКОРОЧЕННЯ.....	3
ВСТУП.....	4
РОЗДІЛ І. ДИФЕРЕНЦІЙОВАНИЙ ПІДХІД В НАВЧАННІ УЧНІВ	6
1.1 Сутність диференційованого підходу в освітньому процесі.....	6
1.1.1 Рівнева диференціація при теоретичному навчанні інформатики..	11
1.1.2 Рівнева диференціація при практичному навчанні інформатики ...	17
1.2 Диференційований підхід і дистанційна форма навчання	21
Висновки до розділу І	23
РОЗДІЛ ІІ. ДИФЕРЕНЦІЙОВАНИЙ ПІДХІД ПРИ НАВЧАННІ УЧНІВ ПРОГРАМУВАННЮ	25
2.1 Огляд програм та підручників з інформатики в контексті вивчення програмування при профільній та рівневій диференціації	25
2.2 Методичні рекомендації та авторські завдання для диференційованого підходу учнів при навчанні інформатики.....	30
2.3 Практикум з програмування.....	37
Висновки до розділу ІІ.....	44
ВИСНОВКИ	45
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	48

СКОРОЧЕННЯ

ІІІ – штучний інтелект

AI - artificial intelligence (штучний інтелект англійською)

GPT - Generative Pre-trained Transformer

ЗЗСО – заклади загальної середньої освіти

МОН – Міністерство освіти та науки України

ЗФПО – заклади фахової передвищої освіти

fizmat@sspi.edu.ua
суворо дотримуйтесь
правил академічної
добросовісності

ВСТУП

Актуальність дослідження застосування диференційованого підходу при вивченні програмування в школі зумовлена низкою сучасних освітніх і соціальних чинників. По-перше, розвиток інформаційних технологій та цифрової економіки вимагає від освітніх систем формування у школярів не лише базових навичок роботи з комп'ютером, а й умінь алгоритмічного мислення, розробки програмних продуктів і критичного аналізу даних. Програмування стає не просто навчальною дисципліною, а ключовою компетенцією XXI століття, інтегрованою у STEM-освіту та міжпредметні проєкти.

По-друге, класична модель викладання програмування часто не враховує різний рівень підготовки учнів, їхні когнітивні здібності, темпи засвоєння матеріалу та індивідуальні стилі мислення. Це створює ризик пасивного засвоєння знань частиною учнів та знижує мотивацію до навчання. У цьому контексті диференційований підхід дозволяє організувати навчальний процес так, щоб завдання, методи і темп роботи відповідали конкретним освітнім можливостям кожного учня.

По-третє, застосування диференціації у навчанні програмуванню сприяє формуванню індивідуальних освітніх траєкторій, що забезпечує як засвоєння базових компетентностей, так і розвиток здібностей учнів із високим рівнем підготовки. Це особливо актуально у шкільній практиці, де змістові та рівневі відмінності учнів потребують адаптивних методик навчання.

Метою дослідження було визначено методичних особливостей застосування диференційованого підходу в навчанні програмуванню учнів середньої школи та розробка рекомендацій щодо оптимізації освітнього процесу з урахуванням індивідуальних потреб учнів з урахуванням особливостей змішаного навчання.

Об'єктом дослідження виступає освітній процес у школі, зокрема організація навчальної діяльності з програмування.

Предметом дослідження є диференційований підхід до організації навчальних занять з програмування, а також їхній вплив на навчальні досягнення, розвиток пізнавальних здібностей та мотивацію учнів.

Мета, об'єкт та предмет дослідження визначають **завдання**:

- 1 Проаналізувати сучасні підходи до організації диференційованого підходу при вивченні програмування.
- 2 Проаналізувати сучасні наукові погляди на поняття та реалізацію диференційованого підходу, зокрема при навчанні програмуванню.
- 3 Розробити завдання для уроків при вивченні програмування.
- 4 Скомпонувати завдання для проведення практикуму з програмування.
- 5 Систематизувати та узагальнити методичні рекомендації реалізації диференційованого підходу при вивченні програмування учнями старших класів.
- 6 Впровадити під час виробничої педагогічної практики розроблені авторську програму та опрацювати отримані дані.

Практична значущість роботи полягає в розробленні навчальних матеріалів для уроків з інформатики при вивченні програмування, підбору завдань для практикуму та узагальнення методичних рекомендацій до організації диференційованого підходу при навчанні програмуванню.

Апробація результатів дослідження відбувалася під час проходження практики.

Структура роботи. Кваліфікаційна робота складається зі вступу, двох розділів, висновків і списку використаних джерел. Загальний обсяг роботи 50 сторінок. Наочність матеріалу досягається використанням 4 рисунки, формуванням 1 таблиця. Опрацьовано 31 джерело.

РОЗДІЛ І. ДИФЕРЕНЦІЙОВАНИЙ ПІДХІД В НАВЧАННІ УЧНІВ

1.1 Сутність диференційованого підходу в освітньому процесі

Метою освітнього процесу в закладах загальної освіти є розвиток учня та всіх його здібностей. Так, в українському законодавстві, зокрема, в Законі про освіту зазначається, що “метою освіти є всебічний розвиток людини як особистості та найвищої цінності суспільства, її талантів, інтелектуальних, творчих і фізичних здібностей...” [5]. В Законі про повну загальну середню освіту уточнюється до вищезазначеного, що особистість має уміти піклуватися про природу, бути готовою до самовдосконалення, свідомого вибору власного майбутнього, виявляти громадянську активність, дбайливо ставитися до родини, країни, довкілля тощо [6].

Там же виокремлюються і деталізуються ключові цілі освіти, а саме [6]:

- всебічний розвиток, виховання і соціалізація учня;
- практична спрямованість освіти для формування знань та умінь для повсякденного життя;
- розвиток критичного мислення особи, яка навчається, та її навичок вирішувати проблеми;
- усвідомлення учнем цінностей свободи, прав людини;
- створення умов для самовдосконалення здобувача;
- створення умов для усвідомленого вибору майбутньої професії та самореалізації;
- розвиток готовності бути активним в глобалізованому світі, толерантним, уміти комунікувати.

Тому важливою є реалізація диференційованого підходу. Такого освітнього процесу, де кожен учень має час і можливості визначити свої здібності, свої запити, чітко сформулювати їх для себе, розуміти за якими критеріями визначити власний рівень результатів навчальних досягнень.

Диференційований підхід у шкільній освіті розглядається як організація навчального процесу, що передбачає цілеспрямоване врахування індивідуальних відмінностей учнів. Йдеться про варіативність змісту, методів, форм і темпу навчання відповідно до рівня підготовки, пізнавальних можливостей, особливостей мислення та темпу засвоєння матеріалу. Такий підхід дає змогу оптимізувати освітній процес, забезпечуючи доступність навчального матеріалу для всіх здобувачів освіти, водночас створюючи умови для розвитку учнів з вищим навчальним потенціалом. У межах диференційованого навчання педагог добирає завдання різного рівня складності, комбінує індивідуальні та групові форми роботи, поєднує різні способи подання інформації. Це сприяє більш точному врахуванню навчальних потреб школярів та підвищує ефективність засвоєння знань. Диференціація також спрямована на підтримання внутрішньої мотивації учнів, оскільки кожен отримує завдання, що відповідають його можливостям і водночас стимулюють подальший розвиток.

Застосування диференційованого підходу розглядається як необхідна умова якісної освіти, оскільки забезпечує індивідуалізацію навчального процесу та сприяє формуванню стійких освітніх результатів у різних групах учнів.

Для з'ясування змісту диференційованого навчання потрібне уточнення основних термінів, пов'язаних із явищем диференціації. Сам термін диференціація (від лат. *differentia* - «відмінність») трактується як поділ цілого на частини, розмежування елементів за певними ознаками або виокремлення різних рівнів і форм. У науковій літературі розрізняють такі поняття: диференціація навчання як умова функціонування освітньої системи; диференційоване навчання як специфіка організації навчального процесу; диференційований підхід як спосіб побудови діяльності вчителя та взаємодії з учнями [2, С. 17; 7].

У контексті сучасної особистісно орієнтованої освіти інтерес до диференціації значно зріс, хоча сама ідея має тривалу історію практичного

застосування, що відображено в аналізі розвитку цього напрямку, здійсненому Н. Дічек [2; 3]. Ретроспективний огляд педагогічної думки XX століття демонструє еволюцію поглядів: у 1960-х роках увага науковців була зосереджена на особистісному підході та пошуках ефективних шляхів урахування індивідуальних особливостей учнів. У 1970-х роках здійснювалися спроби реалізації індивідуального й диференційованого підходів, зокрема через організацію самостійної діяльності школярів і їх групування за певними характеристиками.

У 1980-х роках дослідження були спрямовані на теоретичне обґрунтування принципів індивідуалізації та диференціації, формулювання дидактичних вимог до побудови навчального процесу з урахуванням типологічних відмінностей учнів. У цей період сформовано концептуальні засади індивідуалізованого й диференційованого навчання, що описували структуру й логіку організації цілісної навчальної діяльності. У 1990-х роках акцент було зміщено на необхідність розроблення технологій, здатних забезпечити практичну реалізацію індивідуалізованого та диференційованого навчання в освітній практиці.

Після простеження розвитку диференційованого навчання в педагогічній теорії та практиці логічним кроком є визначення того, яких цілей і завдань прагнули досягти науковці, упроваджуючи цю концепцію в освітній процес.

Соціальний вимір. Диференціація навчання розглядається як засіб формування інтелектуального, творчого та професійного потенціалу суспільства, що забезпечує більш повне й ефективне використання можливостей кожної особистості (Ю. Бабанський, М. Данилов, І. Осмоловська, М. Шахмаєв та ін.).

Психологічний аспект. Метою диференціації є створення умов для переходу до суб'єкт-суб'єктної моделі взаємодії між учнем і вчителем, за якої кожен учасник освітнього процесу стає чинником розвитку іншого. Основою таких відносин виступає спільна продуктивна діяльність, що сприяє

саморозвитку учнів і педагогів (Б. Ананьєв, Л. Виготський, П. Гусак, Л. Занков, А. Палій та ін.).

Дидактичний вимір. Завдання диференціації полягає у створенні сприятливих умов для виявлення задатків і розвитку здібностей кожної дитини. Це досягається завдяки добору різноманітних методів, форм і засобів навчання, які враховують індивідуальні особливості учнів та запобігають нівелюванню їхньої особистості. Ключовим механізмом реалізації цього підходу є надання можливості вибору (Г. Васьківська, Л. Вороніна, В. Галузинський, М. Євтух, В. Кизенко, В. Онищук, В. Паламарчук, О. Савченко, С. Трубачева, О. Ярошенко та ін.).

Методичний підхід. Диференційоване навчання розглядається як інструмент розв'язання актуальних проблем загальної середньої освіти шляхом створення більш ефективної методичної системи. Вона ґрунтується на поєднанні зовнішньої диференціації (селективної й елективної) та внутрішньої, насамперед рівневої, за умови чіткого дотримання вимог освітніх стандартів (О. Бугайов, Н. Буринська, М. Бурда, О. Корсакова, З. Слєпкань, Т. Хмара, О. Ярошенко та ін.) [2; 3]

У сучасній педагогічній науці проблематика диференційованого навчання охоплює кілька ключових напрямів, що відображають запити школи та тенденції розвитку освіти [13; 19]:

1. Теоретико-методологічні засади диференціації. Активно досліджуються концептуальні основи диференційованого навчання: зміст, принципи, форми внутрішньої й зовнішньої диференціації, співвідношення диференціації та індивідуалізації. Науковці уточнюють понятійний апарат, моделюють структуру диференційованих освітніх середовищ та визначають умови їхньої ефективності.

2. Психолого-педагогічні механізми врахування індивідуальних відмінностей. Вивчаються закономірності пізнавального розвитку школярів, особливості мислення, стилі навчання, рівні навчальної мотивації. Значна увага приділяється тому, як індивідуальні особливості учнів впливають на

вибір методів і темпу навчання, а також на формування суб'єктної позиції учня.

3. Дидактичні моделі та технології диференційованого навчання. Поширені дослідження, спрямовані на розроблення конкретних методик, рівневих та профільних завдань, варіативних програм і навчальних матеріалів. Окремий напрям - створення технологій вибору для учнів, адаптивних систем оцінювання, а також алгоритмів організації роботи різнорівневих груп.

4. Практична реалізація диференціації в умовах цифровізації освіти. У центрі уваги - можливості цифрових платформ, адаптивних навчальних систем, онлайн-курсів та штучного інтелекту. Дослідники аналізують, як цифрові інструменти допомагають автоматизувати вибір завдань, відстежувати рівень прогресу та формувати індивідуальні траєкторії навчання.

5. Підготовка педагогів до роботи в умовах диференційованого навчання. Актуальними є питання формування професійних компетентностей учителя: уміння діагностувати навчальні можливості учнів, конструювати різнорівневі завдання, організовувати взаємодію в диференційованих групах. Досліджується також ефективність програм підвищення кваліфікації та методичного супроводу педагогів.

6. Оцінювання результатів диференційованого навчання. Вивчаються моделі та критерії оцінювання, що враховують індивідуальний прогрес учня, а не лише відповідність загальному стандарту. Актуальним є пошук інструментів, які дозволяють поєднати об'єктивність оцінки з індивідуальністю навчальної траєкторії.

7. Соціальні аспекти та питання освітньої рівності. Дискусії точаться навколо того, як диференціація впливає на соціальну справедливість, доступність освіти, профілізацію старшої школи та створення умов для розвитку учнів із різними потребами й здібностями.

Існує поділ на рівневу диференціацію та профільну. Рівнева диференціація здійснюється за здібностями, швидкістю, успішністю. Профільна диференціація виокремлює нахили учня та його інтереси. І

здійснюється у вигляді загального спрямування усього навчання (математичний клас, поглиблене вивчення інформатики) або зумовлює додаткові заняття за інтересами.

1.1.1 Рівнева диференціація при теоретичному навчанні інформатики

Диференціація в освітньому процесі впроваджується з метою створити оптимальні умови для кожного. Учитель має створити на уроці такі ситуації, приклади, розподіл на групи, щоб кожному з понад 20 учнів було комфортно і цікаво працювати. Це можливо за умови визначення класифікації – а яким чином розділяти учнів? І кжен учитель для себе обирає своє бачення, адже кожен вчитель має автономію на власному уроці щодо методів, засобів та форм роботи.

На форумі вчителів було запропоновано висловитися що таке диференціація і як саме вчитель реалізує її у себе. Були отримані відповіді, як можна розділити на такі напрямки [4]:

- поділ за когнітивним розвитком, а саме за рівнем розвитку інтелекту, виконання розумових операцій, швидкість засвоєння знань;
- поділ за мотиваційною сферою, а саме за тим на що орієнтується учень при навчанні: на знання зарадь знань, на знання заради успіху, оцінок, на знання заради спілкування;
- поділ за особистісними особливостями, а саме за темпераментом, щоб утримувати активність та увагу, за соціальною зрілістю, умінням та готовністю працювати в групі;
- за емоційним станом, тобто за рівнем стресостійкості, можливістю працювати при різних емоційних станах;
- засвоїли/ не засвоїли матеріал цілої теми, тобто учні, які засвоїли матеріал отримують матеріали і завдання для його поглиблення, а ті, хто не засвоїв вище середнього рівня, отримують матеріал та завдання для коригування матеріалу;

- виконання домашнього завдання означає, що учень підготувався / не підготувався до продовження вивчення теми, тому диференціація відбувається за базою знань для цього уроку та його продовження.

Окрім цього диференціація може поділятися за методами, що представлено в таблиці 1.1 [4].

Таблиця 1.1. Методи диференціації

Назва	Сутність поділу	Приклади на уроках інформатики
Диференціація змісту навчання	Одна тема — різні рівні завдань	Тема «Алгоритми»: Учням, які впевнено працюють, пропонується задача зі складною розгалуженою структурою (наприклад, алгоритм вибору тарифу мобільного оператора). Тим, хто потребує підтримки, — простіший лінійний алгоритм (алгоритм розрахунку вартості покупки). Тема «Таблиці Google/Excel»: Сильній групі — завдання з умовними форматуваннями та формулами IF/VLOOKUP. Основній групі —

		прості формули типу SUM, AVERAGE.
Диференціація темпу навчання	Учні рухаються з різною швидкістю, але всі доходять до результату з однаковою глибиною знань	Тема «Створення презентацій»: Хто працює швидко — додає анімації, гіперпосилання, інтерактивні кнопки. Хто повільніше — виконує базову структуру: титульна сторінка, 3–4 слайди, висновки. Тема «Scratch / Програмування»: Частина класу за 10 хвилин робить простий скрипт «кіт рухається». Час від часу додавати завдання на «премію за швидкість» — мінічелендж: «додай фон і звук», «зроби лічильник балів».
Диференціація форм організації навчання	Різні формати роботи для різних дітей	Тема «Кібербезпека»: Ті, кому легше працювати в групах, створюють «пам'ятку безпечного інтернет-користувача» у командах. Ті, хто любить тишу й порядок, готують індивідуальну інфографіку. Тема «Бази даних»: Група працює над спільною базою (дані

		про шкільний гурток). Індивідуальні учні створюють власну маленьку базу: «Моя бібліотека».
Диференціація методів навчання	різні способи подачі – щоб комфортно було кожному	Візуалам під час теми «Комп’ютерна графіка» — демонстрація інструментів Canva/GIMP на екрані. Аудіалам під час «Мережевих технологій» — коротке чітке пояснення з прикладами життєвих ситуацій (як працює маршрутизація Wi-Fi вдома). Кінестетикам у темі «Програмування» — більше розборів завдань зі складання коду, переміщення блоків, створення власного проєкту.

Диференціація під час пояснення матеріалу в курсі інформатики полягає у цілеспрямованому доборі змісту, методів і форм подання інформації відповідно до рівня підготовленості, стилю мислення та індивідуальних навчальних можливостей учнів. Такий підхід забезпечує оптимальні умови для засвоєння складних понять, що характеризують сучасні інформаційні технології, та сприяє формуванню стійких предметних компетентностей.

Види диференціації представлено на рис.1.1 та деталізовано нижче.

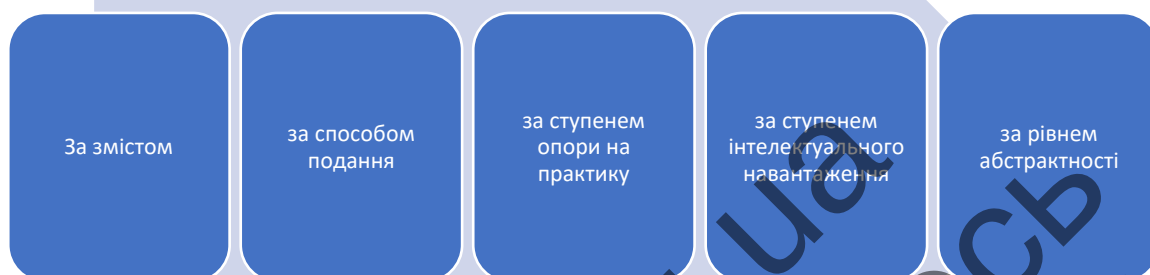


Рис.1.1 Типи диференціації при вивченні теоретичного матеріалу учнями

Диференціація за змістом пояснення. У процесі введення нового матеріалу вчитель змінює глибину та деталізацію теоретичних положень залежно від навчальних можливостей учнів. Наприклад, під час вивчення теми «Комп'ютерні мережі» учням із високим рівнем підготовки подається розширене пояснення про моделі взаємодії мережевих пристроїв, протоколи рівнів OSI та принципи маршрутизації. Учням із базовим рівнем подається спрощена схема будови мережі та приклади її застосування в побуті й школі. Така диференціація дає змогу зберігати єдність теми, але адаптувати обсяг теоретичних знань до можливостей кожної групи.

Диференціація способів подання матеріалу. Оскільки інформатика поєднує абстрактні теоретичні моделі з практичними цифровими інструментами, методична доцільність полягає в комбінуванні різних способів пояснення. Наприклад, у темі «Алгоритми та виконавці» для учнів із наочно-образним мисленням використовуються візуальні блок-схеми, демонстрація руху виконавця в середовищі Scratch або Code.org. Для учнів, які краще сприймають вербальний матеріал, пропонується покрокове пояснення алгоритму у вигляді текстових інструкцій та логічних міркувань. Таким

чином, складний матеріал набуває різних форм подання, що підвищує його доступність.

Диференціація рівня абстракції. У процесі пояснення нових понять учитель змінює рівень узагальнення відповідно до вікових та інтелектуальних особливостей учнів. Наприклад, під час опрацювання теми «Бази даних» учні з високим рівнем теоретичної підготовки опановують поняття «реляційна модель», «первинний ключ», «зв'язки типу один-до-багатьох», тоді як інші учні працюють із прикладними ситуаціями - створенням простої таблиці у середовищі LibreOffice Base або Google Sheets та заповненням її реальними даними. Зміна рівня абстракції дає змогу уникнути перевантаження та підтримати логіку засвоєння матеріалу.

Диференціація за ступенем опори на практику. Матеріал інформатики часто легше засвоюється через практичну діяльність, тому під час пояснення нової теми можна варіювати частку практичних демонстрацій. Наприклад, у темі «Комп'ютерна графіка» для учнів із низьким рівнем навчальної самостійності передбачено детальну демонстрацію дій у графічному редакторі (вибір інструменту, робота з шарами, застосування фільтрів). Учні з вищим рівнем підготовки працюють за короткими інструкціями, здійснюючи самостійний пошук способів розв'язання поставленої задачі та аналізуючи ефективність різних інструментів.

Диференціація за ступенем інтелектуального навантаження. Під час пояснення матеріалу вчитель може давати учням різні за складністю запитання та завдання, що спрямовують їхню пізнавальну діяльність. Наприклад, у темі «Операційні системи» базова група отримує завдання описати призначення робочого столу, файлової структури та основних параметрів Windows. Учні з високим рівнем підготовки аналізують різницю між файловими системами NTFS і FAT32 або пояснюють принципи керування процесами. Це забезпечує участь кожного учня в обговоренні, але з урахуванням його інтелектуального потенціалу.

1.1.2 Рівнева диференціація при практичному навчанні інформатики

Диференціація під час виконання практичних завдань у курсі інформатики має два рівні: первинне закріплення нового матеріалу та підсумкові (контрольні або узагальнювальні) роботи. На кожному етапі диференціація має різну мету й методичне наповнення.

Диференціація *за ступенем складності* передбачає створення системи завдань, структурованих за рівнем когнітивних вимог та операційної складності. На початковому рівні завдання орієнтовані на відтворення базових операцій та застосування засвоєних алгоритмів дій у стандартних умовах. Завдання середнього рівня передбачають необхідність здійснення узагальнень, аналізу та вибору оптимальних способів виконання. Вищий рівень складності характеризується дослідницьким або творчим характером діяльності, що передбачає вміння учня проєктувати власні рішення, модифікувати наявні структури чи оптимізувати процеси.

Такий підхід забезпечує відповідність навчальних вимог реальним можливостям кожного учня, створюючи сприятливі умови для індивідуального розвитку. Розподіл завдань за складністю дає змогу уникати перевантаження слабших учнів і водночас стимулює інтелектуальний ріст сильніших. Диференційовані завдання сприяють розвитку мислення різних рівнів - від репродуктивного до продуктивного та дослідницького. Таким чином формується багаторівнева система навчання, здатна забезпечити досягнення кожним учнем власної освітньої траєкторії.

Учитель пропонує набір практичних вправ, що відрізняються глибиною та кількістю кроків. Приклад: Тема «Табличний процесор». Базовий рівень: створити таблицю та обчислити суму за допомогою функції SUM. Середній рівень: додати формули для обчислення середнього значення та відсоткових показників. Високий рівень: застосувати умовне форматування та створити просту діаграму. Або при іншій темі «Алгоритмічні структури». Базова

вправа: скласти лінійний алгоритм у Scratch. Поглиблена вправа: побудувати алгоритм із розгалуженням та лічильником подій.

Диференціація *за ступенем підтримки* передбачає варіювання обсягу та характеру педагогічної допомоги під час виконання практичних завдань. Учитель може надавати докладні інструкції, послідовні підказки або мінімальні орієнтири залежно від рівня сформованості в учнів операційних умінь та здатності до самостійної роботи. Ступінь підтримки визначається рівнем потреби учня у зовнішній регуляції діяльності та виступає механізмом поступового переходу від керованої роботи до автономної.

Такий підхід сприяє розвитку навичок самостійного прийняття рішень, поступово знижуючи залежність учнів від прямого педагогічного супроводу. Оптимально організована підтримка допомагає уникнути дезорієнтації учнів на початкових етапах практичної діяльності та одночасно формує здатність до самоконтролю й оцінювання результатів. Диференціація за підтримкою підвищує ефективність засвоєння матеріалу та створює передумови для формування стійкої навчальної автономії.

Учитель змінює обсяг допомоги: комусь дає покрокові інструкції, комусь - лише загальний опис задачі. Приклади: Учні, які потребують опори, отримують покрокову інструкцію зі скріншотами (наприклад, «натисніть Вставка → Малюнок»). Учні з вищим рівнем - коротке формулювання задачі, що передбачає самостійний вибір інструментів.

Диференціація *темпу виконання* завдань забезпечує можливість працювати у ритмі, що відповідає індивідуальним когнітивним особливостям учнів. Вона враховує різницю у швидкості сприймання інформації, темпі мислення, здатності планувати роботу та контролювати її виконання. Учням надається можливість працювати у зручному для них темпоритмі, що зменшує рівень стресу та сприяє глибшому опануванню навчального матеріалу без тиску часових обмежень.

У ширшому дидактичному контексті диференціація темпу сприяє формуванню позитивної навчальної мотивації та запобігає виникненню

ситуацій, коли учні відчувають нездатність впоратися із завданням через зовнішні часові рамки. Водночас цей підхід дозволяє раціонально організувати діяльність учнів з високим темпом пізнавальної активності, забезпечуючи їх додатковими можливостями для інтелектуального розвитку. Таким чином формуються умови, за яких темп навчальної роботи стає інструментом підтримки, а не бар'єром.

Діти працюють у власному темпі, але з різною кількістю завдань. Приклад: у темі «Комп'ютерна графіка» усі учні виконують базове редагування зображення; швидкі учні отримують додаткове творче завдання: створити колаж або застосувати фільтри.

Диференціація за формою організації передбачає добір такого типу взаємодії учнів з навчальним матеріалом, який відповідає їхнім індивідуальним особливостям. Учитель може організовувати роботу індивідуально, у парах, малих групах або у форматі змішаних команд залежно від рівня самостійності, комунікативних умінь, навчальних стилів та потреб учнів. Така гнучкість дає змогу ефективно розподілити навчальні ролі та створити умови для оптимальної участі кожного учня у практичній діяльності.

Організаційна диференціація сприяє формуванню соціальних і когнітивних навичок, а також підтримує учнів з різними рівнями академічної підготовки. Вона забезпечує можливість оптимального включення в роботу як тих, хто потребує індивідуалізованих умов, так і тих, хто демонструє високий рівень готовності до колективної взаємодії. Завдяки цьому форма організації не є лише зовнішньою структурою, а виступає важливим елементом педагогічної стратегії, що підсилює навчальну ефективність практичної діяльності. На етапі первинного закріплення учням дають можливість обирати комфортний формат роботи. Приклади: робота в парі для тих, хто відчувається невпевнено; індивідуальне виконання для тих, хто працює швидко та самостійно.

Диференціація під час підсумкових практичних робіт реалізує мету - перевірити рівень сформованості вмінь. Завдання мають бути збалансовані,

але містити елементи диференціації, щоб коректно оцінити як базовий, так і високий рівень. Диференціація рівнів досягнень (модульна побудова): підсумкова робота містить обов'язковий блок і додаткові завдання підвищеної складності. Прикладом може бути створення презентацій. Обов'язковий блок: створити презентацію з 5 слайдів, оформити заголовки, вставити зображення. Поглиблений рівень: додати гіперпосилання, анімацію об'єктів, створити інтерактивний зміст. Іншим прикладом може бути мема «Програмування»: обов'язкове завдання - реалізувати алгоритм із розгалуженням, вищий рівень - реалізувати алгоритм, де поєднані цикли і розгалуження, або оптимізувати вже створений код.

Диференціація за типом завдань при підсумковому контролі може відбуватися з завданнями різного характеру, що дозволяє кожному учню продемонструвати сильні сторони. Практико-орієнтовані завдання - побудувати діаграму за статистичними даними або аналітичні завдання - пояснити, чому використана формула є оптимальною. Також творчі завдання демонструють уміння оформити інформаційний буклет або створити власний макет вебсторінки у візуальному редакторі.

При диференціації за умовами виконання у підсумкових роботах допускається вибір інструментів, що також є формою диференціації. Учень може виконати завдання з графіки в GIMP або у Photopea; роботу з таблицями - у Excel або Google Sheets. Це дозволяє учням працювати у середовищі, у якому вони відчуваються компетентно, не знижуючи при цьому вимог до змісту.

Диференціація за обсягом додаткових завдань при узагальненні знань та демонстрації умінь та навичок доцільно давати учням з різним темпом роботи. Учні, які виконали обов'язкову частину, можуть отримати необов'язкові творчі завдання - як спосіб продемонструвати поглиблені вміння. Наприклад, створити макрос у табличному процесорі; реалізувати нескладну гру у Scratch; створити адаптивний макет вебсторінки з використанням CSS (для старших класів).

1.2 Диференційований підхід і дистанційна форма навчання

Застосування диференційованого навчання в умовах дистанційної форми супроводжується низкою об'єктивних і суб'єктивних труднощів, які впливають на якість освітнього процесу. Насамперед виклики пов'язані з обмеженою можливістю оперативно оцінювати навчальні потреби учнів, що ускладнює коректний добір рівневих завдань і форм взаємодії. Дистанційний формат позбавляє вчителя повноцінного невербального зворотного зв'язку, а отже – зменшує точність визначення темпу роботи, рівня залученості та ступеня розуміння нового матеріалу. Це створює ризик недостатнього або, навпаки, надмірного ускладнення завдань під час диференціації.

Другою групою проблем є неоднакові технічні та організаційні можливості учнів. Різний рівень доступу до техніки, нестабільний інтернет-зв'язок, різна швидкість роботи на цифрових платформах та нерівномірний досвід користування програмним забезпеченням значно ускладнюють реалізацію диференційованих завдань. Утрата оперативності під час виконання практичної діяльності, затримки у передачі матеріалів, а також технічні збої знижують ефективність рівневих інструкцій та індивідуальних маршрутів.

Суттєвою складністю є також потреба у значно більшому обсязі підготовчої роботи з боку вчителя. Створення різнорівневих матеріалів для дистанційного формату потребує не лише педагогічної, а й технічної компетентності: володіння платформами, вміння структурувати матеріали, налаштовувати інтерактивні середовища, опрацьовувати форми онлайн-контролю. Усе це збільшує навантаження та потребує ретельнішої організації часу. За дистанційної форми також зростає потреба у чітких письмових інструкціях, оскільки усна підтримка є обмеженою.

Окремі труднощі стосуються саморегуляції учнів. Дистанційне середовище вимагає вміння самотійно планувати роботу, дотримуватися

термінів та контролювати якість виконаних завдань. Проте учні з низьким рівнем навчальної самостійності виникають у ситуації, коли диференційовані завдання стають для них ще складнішими через відсутність постійного супроводу вчителя. Таким чином виникає ризик збільшення розриву між сильнішими та слабшими учнями, що суперечить меті диференціації.

Загалом дистанційна форма створює умови, у яких диференціація зберігає свою актуальність, але потребує додаткових ресурсів, гнучкості та продуманих організаційних рішень. Вона стає не лише дидактичним підходом, а й складною системою підтримки різних категорій учнів, що потребує чітко вибудованої комунікації та стабільного технічного забезпечення.

Диференційоване навчання в умовах дистанційної форми має низку суттєвих переваг, які підвищують ефективність освітнього процесу та сприяють розвитку індивідуальних навчальних траєкторій учнів. По-перше, дистанційне середовище дозволяє гнучко організовувати навчальні маршрути відповідно до рівня підготовки та темпу засвоєння матеріалу. Використання цифрових платформ дає можливість надавати учням різнорівневі завдання, інтерактивні тести, відеоуроки та презентації, що відповідають їхнім індивідуальним потребам. Завдяки цьому кожен учень може працювати у власному ритмі, глибше засвоювати матеріал і поступово переходити від базових до більш складних завдань.

По-друге, дистанційна форма забезпечує широкі можливості для варіативності методів подання інформації та засобів контролю навчальної діяльності. Учителі можуть комбінувати текстові матеріали, відеодемонстрації, інтерактивні симуляції та онлайн-тести, що відповідають різним стилям навчання учнів. Така різноманітність сприяє кращому засвоєнню складних понять та формує гнучкі компетентності, включно з інформаційною та цифровою грамотністю.

Третя перевага полягає у підвищенні рівня самостійності та відповідальності учнів за власне навчання. Дистанційне середовище створює умови для розвитку саморегуляції, планування часу та виконання завдань за

індивідуальним графіком. Диференційовані завдання мотивують учнів до активного вибору способів і темпу навчання, що формує навички критичного мислення та організації власної діяльності.

Окремо слід відзначити можливість ефективного використання технологічних інструментів для оцінювання та зворотного зв'язку. Цифрові платформи дозволяють оперативно збирати результати виконання завдань, проводити автоматизовану перевірку тестів та анкетування, а також відстежувати прогрес кожного учня. Це дає змогу коригувати навчальний процес у реальному часі та забезпечує індивідуальний підхід до кожного учня без значного збільшення навантаження на вчителя.

Таким чином, дистанційна форма навчання не лише зберігає можливості диференціації, але й розширює їх, надаючи учням та вчителям нові інструменти для ефективної організації навчальної діяльності, підвищення мотивації та розвитку індивідуальних компетентностей.

Висновки до розділу I

Диференційоване навчання виступає ефективним інструментом організації освітнього процесу, що передбачає системне врахування індивідуальних особливостей учнів. Воно забезпечує адаптацію змісту, методів, форм та темпу навчання до рівня підготовки, пізнавальних можливостей, стилю мислення та темпу засвоєння матеріалу. Такий підхід сприяє створенню умов для ефективного засвоєння знань усіма учнями, незалежно від їхнього навчального потенціалу, а також стимулює розвиток здібностей учнів з високим рівнем підготовки.

В освітній практиці виділяють два основні види диференціації: рівневу та профільну. Рівнева диференціація орієнтована на варіації здібностей, швидкості навчання та успішності учнів, що дозволяє формувати індивідуальні навчальні траєкторії та забезпечувати оптимальне засвоєння матеріалу на різних рівнях складності. Профільна диференціація враховує

інтереси та схильності учнів, що проявляється як у загальному спрямуванні навчання (наприклад, математичний або інформаційно-технологічний профіль), так і у наданні додаткових можливостей для розвитку спеціалізованих компетентностей за інтересами.

Застосування диференційованого підходу підвищує ефективність навчання, забезпечує оптимальне співвідношення між рівнем складності завдань і можливостями учнів та сприяє розвитку пізнавальної самостійності. Крім того, такий підхід формує мотивацію до навчання, стимулює творчість і розвиток критичного мислення.

Диференційований підхід є ключовим чинником забезпечення інклюзивності та персоналізації освіти, оскільки дозволяє враховувати як загальні потреби класу, так і індивідуальні особливості кожного учня. Його реалізація сприяє формуванню гнучкої, адаптивної освітньої системи, здатної підвищувати якість навчання, розширювати освітні можливості та забезпечувати комплексний розвиток особистості учня в умовах різнорівневого та профільного навчання.

РОЗДІЛ II. ДИФЕРЕНЦІЙОВАНИЙ ПІДХІД ПРИ НАВЧАННІ УЧНІВ ПРОГРАМУВАННЮ

2.1 Огляд програм та підручників з інформатики в контексті вивчення програмування при профільній та рівневій диференціації

У школах України профільна диференціація відбувається у вигляді розподілу спрямованості класу. Бувають класи з поглибленим вивченням математики, іноземної мови, природничих дисциплін та класи загального спрямування. За назвою зрозуміло що саме вивчають поглиблено учні в даних класах. Дослідивши різні сайти шкіл, нами не було знайдено згадування саме про класи з поглибленим вивченням інформатики, «інформатичних» класів чи класів з поглибленого вивчення цифрових технологій [11]. В школах та ліцеях з переліку зустрічаються переважно математичні класи. Робота, що проводиться з учнями з інформатики, відбувається у вигляді додаткових позаурочних заходів. При цьому наявні підручники для профільного рівня з інформатики [9] та програми для поглибленого вивчення інформатики [12].

Процитуюмо зміст навчання, що відображено в програмі за 2018 рік:

«10 клас

- Мова програмування та структури даних
- Сучасні інформаційні технології
- Аналіз і візуалізація даних
- Графіка\мультимедіа
- Електронні публікації

11 клас

- Бази даних
- Алгоритми
- Веб-технології
- Парадигми та технології програмування» [9, С. 5]

Як бачимо, програмуванню приділяється увага як у 10, так і в 11 класі. Детальніше зміст розбивається на такі теми у 10 класі:

«Мова програмування. Класифікація та складові мов програмування. Особливості середовища розробки. Структура програмного проекту. Основні елементи мови програмування. Використання змінних і виразів. Реалізація базових алгоритмічних конструкцій.

Логічні вирази. Таблиці істинності. Функції. Параметри. Поняття рекурсії. Рекурсивні функції. Вказівники. Поняття структур даних, масив, список, словник, стек, черга, хеш-таблиця. Класифікація структур даних. Лінійні структури даних. Способи реалізації структур даних. Застосування на практиці різних структур даних.

Бібліотеки мови програмування. Створення, редагування та тестування консольних програм і програм з графічним інтерфейсом. Елементи об'єктно-орієнтованого програмування. Правила написання читабельного коду. Коментарі у тексті програми» [9, с. 6]

А в 11 класі програмування розглядається розкриттям таких тем:

«Підходи до системного аналізу, етапи та методології розробки. Уніфікований процес розробки програмного забезпечення.

Інструменти для проектної роботи, системи комунікації та контролю версій.

Мова візуального моделювання архітектури програмного забезпечення .

Аналіз та документація вимог проекту. Діаграми прецедентів.

Моделювання даних і архітектури ПЗ. Діаграми класів.

Моделювання процесів. Діаграми діяльності і послідовностей.

Проектування інтерфейсу користувача. Продуктовий дизайн.

Розроблення прототипу та тестування. Оцінювання системи.

Системна архітектура, апаратні та програмні рішення, стандарти та тренди» [9, С. 12-13].

Спостерігаємо зміст навчання програмуванню поглиблене і воно вимагає часу та зусиль, що зі сторони вчителя, що зі сторони учня.

Іншим способом профільної диференціації саме за темою вивчення в 10-11 класів є підхід до вивчення інформатики з вибілковими модулями. За рівнем стандарту учні в 10 класі вивчають загальні теми повторення за 5-9 клас з поглибленням знань з інформаційних технологій, баз даних, моделювання та роботи зі статистичними функціями. В 11 класі учитель може запропонувати учням обрати один з 11 модулів, серед яких також є модуль «Креативне програмування», присвячений програмуванню [9], в якому пропонується вивчення таких тем:

«Цифрове мистецтво. Дизайн та код. Генеративне мистецтво. Програмування як середовище для творчості. Мова програмування. Особливості середовища розробки. Структура програмного проекту. Основні елементи мови програмування.

Використання змінних і виразів. Реалізація базових алгоритмічних конструкцій. Правила написання читабельного коду. Коментарі у тексті програми. Полотно. Пікселі. Координати. Кольори. Основні форми: точка, лінія, прямокутник, еліпс. Інтерактивність. Обробка подій: переміщення та клацання миші, натиснення клавіш тощо.

Модульність. Функції. Передавання значень у функцію та з неї. Формальні та фактичні параметри. Рекурсії. Фрактали.

Клас, властивості, конструктори, методи. Об'єкти. Події та обробники подій. Взаємодія об'єктів.

Поняття, реалізація та застосування масивів. Текстові рядки як масиви символів. Зображення як цілісний об'єкт та як масив пікселів. Відео як цілісний об'єкт та як масив зображень. Анімації. Трансформації та моделювання руху. Бібліотеки для роботи з мультимедійними даними.

Зовнішні джерела даних. Дані сенсорів та датчиків. Прикладний програмний інтерфейс. Графічний інтерфейс користувача. Поняття та приклади інтерактивних інсталяцій.» [9, С. 62]

При вибіркового опитуванні шкіл м. Суми та Сумської області, м. Полтави та Полтавської області нами не було знайдено учителів, які б обрали

програму профільного рівня чи модуль з вивчення креативного програмування. Було усно опитано учителів понад 20 закладів і з'ясовано, що переважна більшість готує учнів з програмування використовуючи позаурочну роботу, а саме індивідуальні консультації.

У наукових публікаціях про досвід Китайської народної республіки зазначається про досвід рівневої диференціації на рівні класу. У дослідженні [20] показано, що деякі китайські середні школи справді практикують «ability-tracking», що означає ділення учнів на «клас/групу за здібностями»: в одних - «клас для учнів із високими здібностями», в інших - звичайний.

У статті [17] йдеться, що серед освітянських досліджень є і китайські - вони розглядають диференційоване навчання як підхід, що реагує на різні освітні потреби учнів. Згідно з аналізом 2023 року, у багатьох китайських школах учителі й адміністрації так чи інакше інвестують більше зусиль у «талановитих» чи «обдарованих» дітей. Хоч формально офіційних політик «трекінгу» як єдиної системи може й не бути - практика часто прихована, з огляду на те, що «відстежування здібностей» у школах 2000-х років офіційно було заборонене. Дослідження [15; 17; 18; 20] показують, що навіть серед шкіл, які застосовують групування за здібностями, ефект на результати (зокрема з математики), самооцінку учнів чи їхній рівень тривожності - неоднозначний.

Така диференціація пов'язана із спробою «раціоналізувати» освітній процес: виділити сильніших, мотивованіших або здібніших, щоб ефективніше працювати з ними; з можливістю пропонувати різні освітні траєкторії - з поглибленим навчанням, факультативами, або, навпаки, адаптованим темпом для слабших учнів. В реальності відбувається, що деякі учні потрапляють до «класів для сильніших» - з вищими вимогами або більшим навантаженням, а інші - у стандартні класи, де навчання може бути повільнішим або менш інтенсивним. Така система - не є зразком всієї країни: усе залежить від школи, регіону, ресурсу, вчителів. Проте факти таких досліджень є **неоднозначні**:

деякі дослідження показують користь для сильних учнів (краща підтримка), інші - відсутність або негативні соціальні наслідки для слабших.

В дослідженнях українських науковців про шкільну систему Китаю зустрічаємо таке бачення [10;14]. Вислів Конфуція: «Учіться так, ніби ви постійно відчуваєте нестачу своїх знань, і так, ніби ви постійно боїтеся розгубити свої знання» адекватно відображає особливості освітньої системи Китаю, яка є однією з найбільших у світі за чисельністю учнів, кількістю навчальних програм, класів та навчальних закладів.

Освітня система Китаю орієнтована на досягнення високих академічних результатів, причому суттєву роль у цьому відіграє державна підтримка та увага до освітньої сфери. Школи у країні мають різний статус і рівень престижності: експериментальні заклади при університетах та приватні навчальні заклади вважаються найбільш престижними, тоді як навчальні заклади сільської місцевості характеризуються нижчим рівнем ресурсного забезпечення. Це зумовлює міграцію учнів із сільських районів до міст із подальшим проживанням у гуртожитках.

Незважаючи на те, що всі школи отримують державне фінансування на оновлення матеріально-технічної бази та ремонт приміщень, освіта в Китаї не є повністю безоплатною. Учні та їхні родини сплачують абонентську плату за семестр, яка в невеликих містах становить приблизно 400 юанів (майже 1600 грн), а також додаткові витрати на шкільну форму, підручники та харчування. Внаслідок цього витрати на освіту можуть складати до третини сімейного бюджету.

Структурно шкільна освіта Китаю передбачає 12-річне навчання. Початкова школа охоплює перші шість років навчання, середня - наступні три, а старша - останні три роки. Обов'язковою є дев'ятирічна освіта, після завершення якої рішення щодо продовження навчання у старших класах приймають батьки та учень. Конкуренція у китайській освіті є значною. Для вступу до першого класу, переходу до старшої школи або вищого навчального закладу учні зобов'язані скласти відповідні іспити та проходити конкурси.

Зарахування до початкової школи здійснюється за територіальним принципом.

Існують також школи при університетах, наприклад, при Zhejiang Normal University, до яких вступ майже завжди гарантує подальше зарахування у вищі навчальні заклади для учнів, що набрали достатню кількість балів.

Вступ до університетів великих міст, таких як Пекін або Шанхай, є висококонкурентним: на одне місце в престижних навчальних закладах може претендувати від 200 до 300 осіб. Водночас обдаровані учні користуються державними пільгами, серед яких стипендії та субсидії підприємств і організацій, що сприяє їхньому просуванню освітніми рівнями.

Таким чином, диференціація профільна ні в Україні, ні в Китайській народній республіці не реалізована у повній мірі. Натомість рівнева диференціація має певне місце в шкільній освіті.

2.2 Методичні рекомендації та авторські завдання для диференційованого підходу учнів при навчанні інформатики

Загальні принципи диференційованого навчання полягають у певних правилах його реалізації. Це стосується організації усвідомленого підходу учнів до диференціації, їх залучення до такого процесу, що полягає у наступному:

1. Врахування рівня підготовки учнів – розподіл завдань і активностей за рівнем складності: базовий, середній, підвищений.
2. Індивідуальні освітні траєкторії – надання можливості учням обирати темпи опрацювання матеріалу та вид завдань відповідно до власних інтересів і сильних сторін.
3. Різні форми подачі матеріалу – використання лекцій, інтерактивних демонстрацій, відеоуроків, програмних симуляторів, наочних схем алгоритмів.

4. Акцент на компетентнісний підхід – не лише знання синтаксису, а й уміння застосовувати алгоритмічне мислення, вирішувати практичні задачі та аналізувати результати.

Структура уроків програмування з диференціацією може бути такою:

- Вступна частина (5–10 хв)
- Ознайомлення з новою темою через приклади різного рівня складності.
- Короткий контроль наявних знань через опитування або онлайн-тести.
- Основна частина (25–30 хв)
- Базова група: робота над простими алгоритмічними завданнями, практичне закріплення теми.
- Середня група: виконання завдань із застосуванням умов, циклів, функцій, структур даних.
- Підвищена група: складні задачі на оптимізацію алгоритмів, робота з бібліотеками, розв'язування кейсів із реального програмування.
- Заключна частина (5–10 хв)
- Обговорення результатів, пояснення типових помилок.
- Рефлексія: учні оцінюють власний рівень і прогрес.

Використання різних засобів і методів на уроках також дають змогу реалізувати диференційований підхід. Це можуть бути:

1. Рівневі завдання – одна тема, кілька варіантів завдань за складністю.
2. Індивідуальні або парні проекти – дозволяють учням обирати теми та засоби реалізації.
3. Гнучке групування – тимчасові групи для вирішення конкретних задач, що змінюються залежно від теми та успішності.
4. Менторство старших учнів або більш підготовлених – допомога учням із нижчим рівнем знань через підтримку однолітків.
5. Використання електронних платформ (Code.org, repl.it, PythonTutor, Scratch, Visual Studio Code) – для самостійної роботи та контролю прогресу.

Важливим етапом є організація і проведення контролю та оцінювання.

Формувальне оцінювання це є короткі тести, перевірка коду, онлайн-опитування. Адитивному оцінюванню підлягають підсумкові проекти, програмні завдання, участь у конкурсах та олімпіадах. Також доцільно пропонувати учням досвід у самооцінюванні та взаємооцінюванні.

Зворотний зв'язок учителя та його рекомендації щодо покращення алгоритмів, оптимізації коду, стилю програмування — є важливим етапом реалізації диференційованого підходу.

Велика кількість учителів та методистів завдяки досвіду та науковим дослідженням випрацювали загальні рекомендації до роботи з групами різного рівня підготовки.

1. Варто планувати уроки з урахуванням часу для самостійної роботи різних груп.
2. Необхідним є створювати банк завдань різної складності, який можна постійно оновлювати.
3. Доцільно застосовувати інтерактивні вправи для розвитку алгоритмічного мислення та логіки.
4. Здійснювати заохочення виявляти креативність та дослідницьку діяльність, надаючи учням можливість самостійно обирати проекти.
5. Використовувати електронні засоби для моніторингу прогресу та адаптації завдань у реальному часі.

Було розроблено декілька уроків з завданнями для реалізації диференційованого підходу на тему «Модулі, функції та методи опрацювання числових даних мовою Python» за освітньою програмою для профільного рівня 10 клас.

Нижче наведено приклади завдань для трьох рівнів: середнього, достатнього та високого. Кожен рівень має два завдання. Не зазначаємо початковий рівень, оскільки передбачається, що учні, які навчатимуться у класі за програмою профільного рівня швидко опановують базові знання.

Середній рівень. Мета: закріплення базових знань про модулі, функції та стандартні методи обробки чисел.

Завдання 1. Напишіть функцію `sum_of_squares(n)`, яка обчислює суму квадратів чисел від 1 до n включно. Використайте стандартні математичні операції та цикл.

Завдання 2. За допомогою модуля `math` обчисліть квадратний корінь із суми трьох чисел, що вводяться користувачем. Напишіть функцію `root_sum(a, b, c)`, яка повертає результат.

Достатній рівень. Мета: застосування функцій, робота з модулями, перетворенням числових даних.

Завдання 1. Напишіть функцію `is_prime(n)`, яка перевіряє, чи є число простим. Потім за допомогою цієї функції виведіть усі прості числа в діапазоні від 2 до 50. Використайте модуль `math` для оптимізації перевірки дільників.

Завдання 2. Створіть функцію `round_list(numbers)`, яка приймає список чисел і повертає новий список, де всі числа округлені до найближчого цілого. Використайте метод `round()` з Python.

Високий рівень. Мета: комплексна робота з функціями, модулями, списками, обробкою числових даних та застосування алгоритмічного мислення.

Завдання 1. Напишіть функцію `stats_summary(numbers)`, яка приймає список чисел і повертає словник з наступними статистичними даними: мінімум, максимум, середнє арифметичне, суму квадратів чисел. Використайте модуль `math` для обчислень, а також функції `min()`, `max()` та `sum()`.

Завдання 2. Створіть функцію `factorial_digit_sum(n)`, яка обчислює факторіал числа n (за допомогою модуля `math`) і повертає суму цифр результату. Наприклад, для $n = 5$ факторіал $= 120$, сума цифр $= 1 + 2 + 0 = 3$.

Опишемо розв'язання одного з завдань кожного рівня. Перше завдання: обчислення суми квадратів від 1 до n .

Створюємо функцію з параметром n .

Ініціалізуємо змінну для накопичення суми.

За допомогою циклу for перебираємо числа від 1 до n і додаємо квадрат кожного до суми.

Повертаємо результат.

```
def sum_of_squares(n):  
    total = 0  
    for i in range(1, n+1):  
        total += i**2 # додаємо квадрат числа  
    return total
```

Завдання достатнього рівня про прості числа від 2 до 50.

Функція перевіряє, чи число n є простим:

Якщо число $\leq 1 \rightarrow$ не просте.

Перевіряємо дільники від 2 до $\sqrt{n}$. Якщо є дільник $\rightarrow$ число не просте.

Створюємо цикл від 2 до 50 і виводимо тільки прості числа.

```
import math  
def is_prime(n):  
    if n <= 1:  
        return False  
    for i in range(2, int(math.sqrt(n)) + 1):  
        if n % i == 0:  
            return False  
    return True  
for num in range(2, 51):  
    if is_prime(num):  
        print(num, end=' ')
```

Завдання високого рівня про обчислення мінімального, максимального значення, середнього та суми квадратів.

Функція приймає список numbers.

Обчислюємо min(numbers) і max(numbers).

Обчислюємо середнє: sum(numbers)/len(numbers).

Обчислюємо суму квадратів через цикл або генератор.

Повертаємо словник із результатами.

```
import math

def stats_summary(numbers):

    minimum = min(numbers)

    maximum = max(numbers)

    mean = sum(numbers) / len(numbers)

    sum_squares = sum(x**2 for x in numbers)

    return {

        "min": minimum,

        "max": maximum,

        "mean": mean,

        "sum_squares": sum_squares }
```

Розвиток учнів відбувається поступово у випадку кожного завдання. Завдання середнього рівня розвивають увагу, аналіз, синтез, навички планування та абстрагування. Завдання достатнього рівня додають порівняння, оптимізацію та вміння інтегрувати функції у більш складний алгоритм. Найскладніші завдання формують комплексне мислення: аналітику даних, узагальнення результатів, структуровану подачу інформації, планування багатокрокового алгоритму та оцінювання ефективності.

Тобто при виконанні завдання середнього рівня учень поглиблює вміння аналізувати такими діями: розбиває завдання на логічні кроки: «потрібно знайти квадрати чисел», «потрібно додати їх», виділяє головне (числовий діапазон, операція піднесення до квадрату) і другорядне (ініціалізація змінної для суми).

Розвиток уміння застосовувати синтез відбувається, коли учень об'єднує послідовність операцій у єдину функцію, створює алгоритм вирішення завдання.

Також задіюється вміння абстрагування – учень переходить від конкретних чисел до загальної формули (сума квадратів чисел від 1 до n).

Узагальнення відбувається на етапі розбору з вчителем, а саме при усвідомленні, що ця функція працює для будь-якого натурального числа $n \rightarrow$ формування навички застосування шаблонів.

Учень планує алгоритм: цикл $\rightarrow$ квадрат $\rightarrow$ додавання $\rightarrow$ повернення результату, а значить розвиває алгоритмічний підхід до розв'язування задачі.

При виконанні завдання достатнього рівня учень розділяє завдання на підзадачі: перевірка простоти числа, перебір чисел у діапазоні. Виділяє умови: «число ділиться на інші числа чи ні». Поеднує функцію `is_prime` з циклом для отримання повного списку простих чисел $\rightarrow$ формує комплексний алгоритм. Учень порівнює результати дільників із числом, що перевіряється. Він демонструє або набуває уміння мислити не лише конкретними числами, а й закономірностями простих чисел; аналізує ефективність алгоритму (перевірка дільників до $\sqrt{n} \rightarrow$ оптимізація).

В завданнях високого рівня відбувається спонукання учня до розкладання завдання на підзадачі: пошук мінімуму, максимуму, середнього, суми квадратів; визначення, які функції Python допомагають швидко отримати результат; поєднання окремі обчислення в єдину функцію, яка повертає структурований результат у словнику; подальший розвиток уміння мислити категоріями: «числовий список», «статистика», «структура даних словник» замість конкретних чисел. Врешті учень прогнозує послідовність дій: визначення статистичних величин $\rightarrow$ обчислення $\rightarrow$ зберігання в структурі $\rightarrow$ повернення результату. А також відбувається оцінювання власної роботи, перевірка правильності обчислень, правильності формату результату та оптимізації коду (генератор для суми квадратів).

Таким чином такі завдання розвивають учнів в процесі роботи над розробкою коду і сприяють подальшому набуттю компетентностей у галузі програмування.

2.3 Практикум з програмування

Одним зі способів реалізації рівневої та профільної диференціації є проведення практикуму.

Практикум у сучасній школі розглядається як інтерактивна форма організації навчання, у межах якої здобувачі освіти виконують комплекс прикладних завдань та/або міні-проектів, що потребують активного застосування теоретичних знань й цифрових інструментів. Під час такого заняття учні працюють індивідуально або в малих групах, що дає змогу оптимізувати робочий процес та підтримати активність учасників. Типова структура практикуму включає представлення теми й цілей, актуалізацію попередніх знань, мотиваційний етап, опрацювання інструкцій, виконання завдань, презентацію результатів та їх рефлексивно-аналітичне обговорення.

Участь у практикумі створює умови для розвитку індивідуальних інтересів, самостійності та дослідницького стилю мислення. Формат заняття підтримує диференційований підхід: учні можуть отримувати завдання різної складності - від чітко структурованих інструкцій до відкритих задач, у яких необхідно самостійно сформулювати алгоритм розв'язання, підібрати інструменти та обґрунтувати вибір способів роботи. Особливо продуктивним є застосування цього підходу під час практикумів з програмування, де можлива одночасна робота над різнорівневими задачами, мікропроектами або короткими дослідженнями даних.

Практикум є однією з найскладніших, але й найбільш результативних форм організації самостійної діяльності учнів. Залучення до активної роботи сприяє формуванню навичок планування, постановки дослідницьких питань, використання інструментів цифрового аналізу, створення та тестування алгоритмів, інтерпретації результатів і формування аргументованих висновків. Така діяльність робить можливим індивідуальний темп роботи,

поступове ускладнення завдань і перехід учнів на вищі рівні когнітивної активності.

Підготовка практикуму передбачає визначення його місця в системі уроків, розроблення інструкцій та диференційованих завдань, створення критеріїв оцінювання, забезпечення доступу до необхідного програмного забезпечення та обладнання, підготовку цифрових ресурсів і планування системи педагогічного супроводу - як безпосереднього, так і опосередкованого. Важливо також забезпечити комфортні умови роботи та підтримати творчу атмосферу, що сприяє академічній ініціативності та дотриманню етичних норм взаємодії в групі.

У рамках практикумів із програмування особливо значущими є використання сучасних цифрових середовищ (IDLE, онлайн-платформи для тестування коду, системи візуалізації даних), застосування елементів алгоритмічного мислення та методів роботи з інформацією. Це дає змогу інтегрувати навчальну діяльність із реальними ІТ-практиками та формує основи цифрової компетентності.

Під час аналізу й оцінювання практичних занять і, зокрема, практикуму доцільно звертати увагу на такі аспекти [1]:

- відповідність змісту роботи навчальній програмі та логіці вивчення теми;
- чіткість визначення цілей та очікуваних результатів діяльності учнів;
- якість підготовки матеріалів (зміст інструкцій, актуальність цифрових інструментів, доступність ресурсів);
- організацію процесу виконання завдань, доречність методів інструктажу, своєчасність педагогічної підтримки та корекції;
- результативність роботи учнів: здатність застосовувати знання, розробляти та тестувати алгоритми, аналізувати дані, документувати виконану роботу та обґрунтовувати висновки.

Існує велика кількість платформ для навчання програмуванню. Одним з таких ресурсів є практикум з 828 завданнями у відкритому доступі [8].

З його переліку було сформовано пакет завдань для проведення практикуму. Такий пакет можна використовувати для індивідуального розвитку здобувача в галузі програмування або для проведення змагань. В межах даного дослідження було запропоновано учням 10-11 класів пройти такий практикум. Він розрахований на роботу протягом навчального тижня. Тобто протягом 5 робочих днів учень вдома або в класі закладу освіти міг вирішувати такі завдання. Після виконання усіма учасниками проводилася зустріч і обговорювалися альтернативні варіанти розв'язування, якщо вони були, або розбиралися наявні з дискусією щодо оптимізації коду.

Завдання для практикуму пропонувалися так, що наведені нижче з сервісу [8]

1. Напишіть програму, яка отримує три цілих числа, введені з клавіатури (кожне число вводиться на окремому рядку), і друкує на екрані їх суму, добуток, результат піднесення першого числа до степеня різниці другого і третього чисел.
2. Напишіть програму, яка приймає ім'я та прізвище від користувача у різних рядках та роздруковує їх у зворотному порядку з пропуском між ними у вітальному повідомленні.
3. Дано двоцифрове число. Знайдіть число десятків у ньому.
4. Напишіть програму, в якій користувач вводить пароль і якщо він співпадає із наперед визначеним паролем для цього користувача, то виводиться повідомлення Password accepted.. У іншому випадку повідомлення буде Sorry, that is the wrong password..
5. Користувач вводить дві різних англійські літери в окремих рядках. Напишіть програму, яка виводить повідомлення про місце розташування однієї літери відносно іншої у алфавіті.
6. Напишіть програму, яка підраховує додатні і від'ємні числа, а також нулі, введені користувачем, і виводить їхню кількість в один рядок з одним пропуском між ними.

7. Користувач вводить рядок і набір символів. Напишіть програму, яка перевіряє чи починається рядок із зазначених символів.
8. Збережіть назви мов світу, які вводяться в одному рядку через пропуск, у списку. Простежте за тим, щоб елементи у списку не зберігались в алфавітному порядку. Відсортуйте список в алфавітному порядку і виведіть його елементи в рядку через пропуск.
9. Створіть програму, яка отримує на вхід послідовність цілих чисел, і друкує на екрані: найменше число у списку, найбільше число у списку, кількість чисел у списку, середнє значення елементів у списку.
10. Створіть словник з інформацією про пошукові системи, в якому ключі – назви пошукових систем, а значення – частка їхнього використання у світі на даний момент (дійсне число). Надрукуйте елементи словника як у вихідних даних. Елементи у словнику невпорядковані.
11. Ви створюєте пригодницьку гру і використовуєте для зберігання предметів гравця словник, у якому ключі - це назви предметів, значення - кількість одиниць кожної із речей. Виведіть повідомлення про усі речі гравця як у вихідних даних.
12. Напишіть функцію для перетворення цілого числа в рядок.
13. Напишіть функцію, яка отримує два слова, об'єднує їх за допомогою пропуску і друкує результат об'єднання.
14. Напишіть програму для виведення опису вбудованої функції Python. Імпортуйте модуль `inspect` для розв'язування поставленої задачі. Для прикладу, роздрукуйте опис функції `pow()`.
15. Напишіть програму для перемішування та друку введеної послідовності. Використайте модуль `random`.
16. Напишіть клас, який має як мінімум два методи: перший - отримати рядок з вводу консолі, другий - друкувати рядок у верхньому регістрі.

Завдання розподілялися на 5 днів: 4 дні по 3 завдання і 5 день 4 завдання.

Учні мали змогу виконувати їх у власному темпі. Важливим елементом було обговорення. Завдання не є складними для профільного рівня, проте для

розвитку здібностей в програмуванні це має бути систематичною та постійною роботою.

Регулярна та систематична практика є ключовою умовою підтримання та розвитку професійної компетентності програміста. Її значущість зумовлена низкою закономірностей, що визначають характер діяльності у сфері програмування.

По-перше, алгоритмічне мислення формується та стабілізується лише за умов постійного розв'язання практичних задач. Аналіз проблеми, декомпозиція, вибір структур даних, передбачення можливих помилок — це когнітивні операції, ефективність яких безпосередньо залежить від регулярного тренування.

По-друге, динамічність галузі вимагає постійного оновлення навичок. Програмні засоби, мови та бібліотеки змінюються надзвичайно швидко, тому систематична практика забезпечує актуальність професійних знань і запобігає їх швидкій деградації.

По-третє, якість програмного коду безпосередньо пов'язана з досвідом. Формування навичок написання структурованих, оптимальних і зрозумілих програмних рішень можливе лише через багатократне повторення дій у різних контекстах, що відповідає механізмам набуття професійної майстерності.

По-четверте, регулярна діяльність сприяє розвитку автоматизованих умінь, зокрема швидкого орієнтування в інструментарії, розпізнавання типових патернів і помилок. Ці уміння суттєво підвищують ефективність роботи та знижують когнітивне навантаження.

По-п'яте, постійна практика забезпечує здатність програміста адекватно оцінювати масштаб і складність завдань, прогнозувати ризики та обирати оптимальні стратегії роботи, формуючи професійний досвід на основі накопичених ситуаційних моделей.

По-шосте, систематична діяльність розширює креативний потенціал спеціаліста. Зростання кількості успішно розв'язаних задач сприяє появі більш елегантних, економних і концептуально цілісних рішень.

Нарешті, регулярна практика створює умови для формування портфоліо, підвищує впевненість у власних можливостях і забезпечує поступальний розвиток професійних компетентностей.

Таким чином, систематична практична діяльність є необхідною передумовою професійного зростання програміста, підтримання актуальності його навичок та забезпечення високої результативності роботи в умовах швидко змінюваного цифрового середовища.

Такі завдання були запропоновані учням, які виявили бажання взяти участь в такій діяльності. Кількість таких учнів становила 25 чоловік.

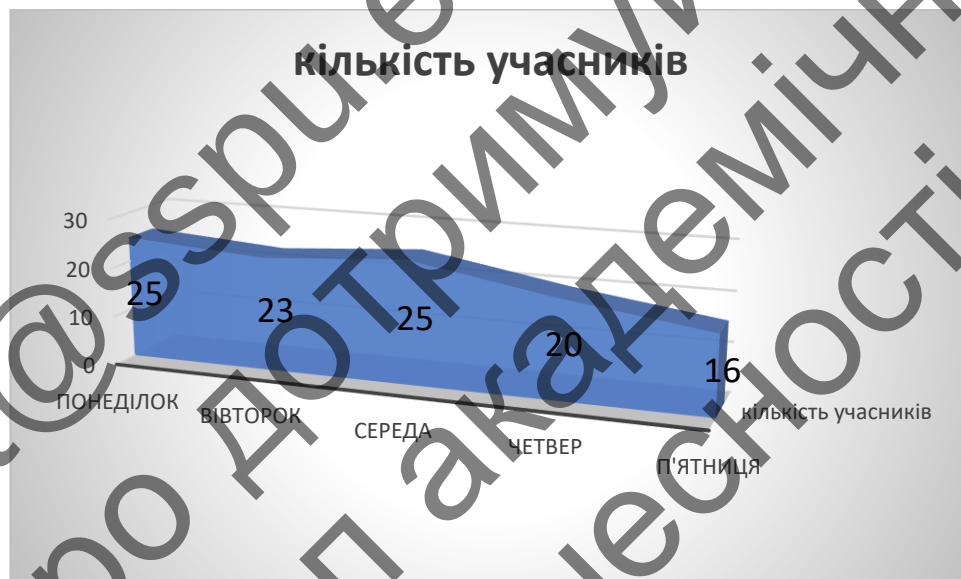


Рис.2.1. Зміна кількості учасників обговорення на зустрічах протягом тижня

На рисунку 2.1 відображено кількість учасників, які брали участь в обговоренні протягом кожного дня онлайн. На рисунку 2.2 відображено кількість правильних розв'язків по кожному з завдань. Учасники могли відмовитися демонструвати власні розв'язування, оскільки участь була добровільною.



Рис.2.2 Кількість правильних розв'язків

На рисунку продемонстровано кількість учасників, які прислали вчителю розв'язки. Навіть, якщо учнів не було на зустрічі, учитель міг провести з ними консультацію в інший день. Це стосується зустрічей у вівторок, четвер і п'ятницю. Наприклад, у п'ятницю на зустрічі були присутні 16 учасників, проте розв'язки до 13 та 14 завдання надіслали 18.

Після проходження такого практикуму учні покращили свої показники з навчання з даної теми. На рис.2.3 відображено порівняння успішності учасників до та після проведення практикуму.

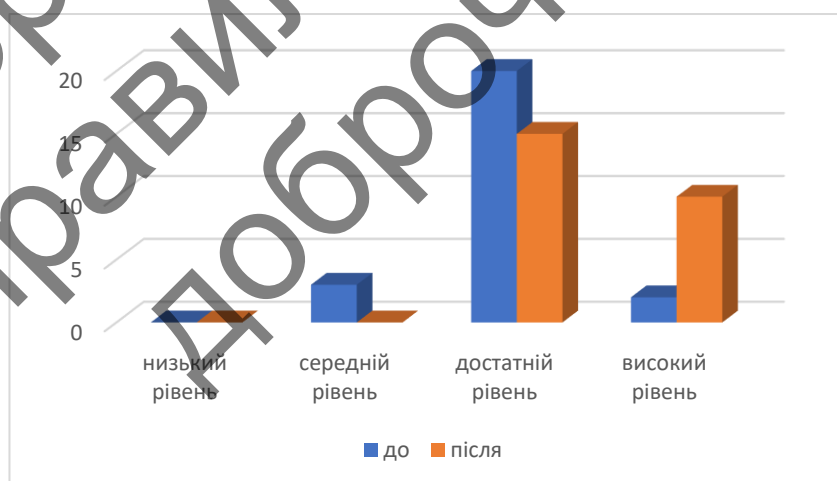


Рис.2.3 порівняння рівнів успішності з програмування учасників практикуму

Висновки до розділу II

Диференціація, що реалізується в закладах загальної середньої освіти, може бути рівневою та профільною. Аналіз програм, підручників, платформ та ресурсів у відкритому доступі глобальної мережі Інтернет дає підстави стверджувати про можливість реалізації обох видів диференціації. Проте опитування декількох шкіл не виявило популярності профільної диференціації з програмування в школах м. Суми та Сумської області. Такий висновок не свідчить про втрату популярності саме програмування в цілому як галузі, а призводить до висновку про необхідність дослідити причини такого вибору в подальшому. Однією з гіпотез може бути необхідність учнів готуватися до національного мультипредметного тесту і небажання витрачати час. Оскільки в контексті даного дослідження даний аспект не було заплановано, то він не вивчався детально.

Рівнева диференціація з програмування застосовується в школах, як і для будь-якої іншої теми.

Для проведення практичної частини дослідження було взято за основу завдання для практикуму з інформатики. Учні мали змогу долучитися добровільно. До практикуму долучалися учні 10-11 класів.

Опрацювання результатів показало, що учні мають бажання працювати з програмуванням. Мова програмування не обмежувалася, оскільки учні були задіяні з різних класів і у них могли викладати різні вчителі. Проте усі програмували мовою Python. Учитель долучався до обговорення та демонстрував раціональний результат розв'язування задач.

ВИСНОВКИ

В роботі проаналізовано теоретичне підґрунтя даної теми, проведена практична частина дослідження та описані результати спілкування з вчителями та здобувачами. Систематизовано матеріал дослідження, що дає підстави сформулювати такі висновки.

1. Диференційований підхід у навчанні розглядається як одна з провідних технологій сучасної освіти, спрямована на узгодження змісту й організації навчальної діяльності з індивідуальними освітніми потребами здобувачів. Його сутність полягає в цілеспрямованій модифікації навчальних завдань, методів і темпу роботи відповідно до реального рівня підготовленості, когнітивних особливостей та інтересів учнів. Завдяки цьому забезпечуються умови, за яких учні з різною навчальною динамікою можуть досягати засвоєння матеріалу на оптимальному для себе рівні, а школярі з високим потенціалом отримують додаткові стимули для інтелектуального розвитку.

2. У теорії та практиці педагогіки найчастіше виокремлюють два ключові вектори диференціації: рівневий та профільний. Рівнева диференціація пов'язана з урахуванням різниці у темпах навчання, стійкості уваги, сформованості базових умінь та навчальних можливостей. Вона дає змогу структурувати навчальні завдання за ступенем складності, забезпечуючи індивідуалізацію освітньої траєкторії. Профільна диференціація, натомість, ґрунтується на виборі учнями змістових напрямів відповідно до їхніх інтересів, мотиваційної сфери та професійних намірів, що дозволяє формувати спеціалізовані компетентності та поглиблювати знання у певній галузі.

3. Застосування системи диференціації справляє комплексний позитивний вплив на освітній процес. Вона забезпечує оптимальний баланс між вимогами навчальної програми та індивідуальними можливостями учнів, стимулює розвиток самостійності у пізнавальній діяльності, підвищує рівень

внутрішньої мотивації та сприяє формуванню критичного, аналітичного та творчого мислення.

4. В умовах переходу до персоналізованої та інклюзивної освіти диференційований підхід виступає ключовим механізмом адаптації навчального середовища. Його впровадження дозволяє створити гнучку систему організації освітнього процесу, у якій водночас враховуються групові особливості класу та індивідуальні потреби кожного здобувача освіти. У результаті забезпечується підвищення якості навчальних результатів, розширення спектра освітніх можливостей та комплексний розвиток особистості учня в умовах багаторівневого та профільного навчання.

5. Загальні рекомендації застосування диференційованого підходу такі:

- Врахування рівня підготовки учнів – розподіл завдань і активностей за рівнем складності: базовий, середній, підвищений.
- Індивідуальні освітні траєкторії – надання можливості учням обирати темпи опрацювання матеріалу та вид завдань відповідно до власних інтересів і сильних сторін.
- Різні форми подачі матеріалу – використання лекцій, інтерактивних демонстрацій, відеоуроків, програмних симуляторів, наочних схем алгоритмів.
- Акцент на компетентнісний підхід – не лише знання синтаксису, а й уміння застосовувати алгоритмічне мислення, вирішувати практичні задачі та аналізувати результати.
- Варто планувати уроки з урахуванням часу для самостійної роботи різних груп.
- Необхідним є створювати банк завдань різної складності, який можна постійно оновлювати.
- Доцільно застосовувати інтерактивні вправи для розвитку алгоритмічного мислення та логіки.
- Здійснювати заохочення виявляти креативність та дослідницьку діяльність, надаючи учням можливість самостійно обирати проекти.

- Використовувати електронні засоби для моніторингу прогресу та адаптації завдань у реальному часі.

6. Були розроблені авторські завдання для проведення уроків при вивченні програмування. А також для зацікавлених в програмуванні учнів був проведений практикум з завданнями з відкритих джерел.

Результати продемонстрували прогрес учнів в програмуванні, їх задоволеність такою формою роботи та зацікавленість в продовженні.

Дане дослідження може бути розширене і продовжене в контексті впровадження практикуму в освітній процес та з'ясування результатів такого впровадження при диференційованому підході навчання програмуванню.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Бондаренко Ю.А. Дослідницький практикум: форма зв'язку теорії з практикою. *На урок*. URL: <https://naurok.com.ua/prezentaciya-na-temu-doslidnickiy-praktikum-forma-zv-yazku-teori-iz-praktikoyu-427052.html>
2. Васьківська Г.О. Генеза диференційованого навчання: дидактичний аспект. Диференціація у шкільній освіті: історичний досвід і сучасні технології: збірник матеріалів Всеукр. науково-практичної конференції, 16 травня 2018 р., м. Київ. - К.: Інститут педагогіки, 2018. С. 16-20.
3. Дічек Н. П. Диференціація навчання як проблема розвитку вітчизняного шкільництва в імперську добу (80-ті рр. XIX ст. - до 1917 р.). *Культурно-історична спадщина Польщі та України як чинник розвитку полікультурної освіти: зб. наук. праць*. - К.: Хмельницький, 2011. - С. 232–241
4. Диференційоване навчання. *Форум. Академія ранок*. URL: <https://surl.li/uxbimu>
5. Закон про освіту. URL: <https://zakon.rada.gov.ua/laws/show/2145-19#Text>.
6. Закон про повну загальну середню освіту. URL: <https://zakon.rada.gov.ua/laws/show/463-20#Text>.
7. Липова Л., Войцехівський М., Замаскіна П., Диференціація як провідний принцип допрофільної підготовки учнів основної школи. 2011. URL: chrome-extension://efaidnbmnnnibpcajpcglclefindmkaj/https://elibrary.kubg.edu.ua/id/eprint/2105/1/L_Lypova_M_Vojtsehivskiy_P_Zamaskina_RSH_1-2_IPPO.pdf
8. Мізюк. О. Практикум з програмування. URL: <https://surl.lu/uqynbc>
9. Освітні програми. МОН України. URL: <https://mon.gov.ua/osvita-2/zagalna-serednya-osvita/osvitni-programi/navchalni-programi-dlya-10-11-klasiv>.
10. Проскура О. Освітні замальовки з Піднебесної: як навчають у Китаї. НУШ: *Нова українська школа*. URL:

<https://nus.org.ua/2018/06/07/osvitni-zamalovky-z-pidnebesnoyi-yak-navchayut-u-kytayi/>

11. Рейтинг шкіл за результатами НМТ. URL: <https://osvita.ua/school/rating/93058/>

12. Руденко В. Д., Речич Н. В., Потієнко В. О. Інформатика (профільний рівень) підручник для 10 класу. URL: <https://pidruchnyk.com.ua/3097-informatyka-10-klas-rudenko.html>

13. Сальник, І. В. Диференціація та інтеграція навчання природничих дисциплін – дві сторони єдиного освітнього процесу. *Наукові записки. Серія: Проблеми природничо-математичної, технологічної та професійної освіти*. С. 48-55. DOI <https://doi.org/10.32782/cusu-pmtp-2023-1-7>.

14. Шпарик О. Китайська шкільна система: сучасний стан та шляхи розвитку. *Український педагогічний журнал*. 2018. URL: <https://uej.undip.org.ua/index.php/journal/article/view/463>

15. Anda Wang. The Impact of The Double Reduction Policy on The Construction of The Professional Identity of Tutoring Class Teachers. *Journal of Education, Humanities and Social Sciences*. March 2024 *Journal of Education Humanities and Social Sciences* 26:29-38. DOI:10.54097/gj768p91.

16. Chen, Dandan; Fu, Ning - GED02; Pan, Yilin. Progress and Challenges of Upper Secondary Education in China (English). Washington, D.C. : World Bank Group. <http://documents.worldbank.org/curated/en/437461554920204978>

17. Nan Zhao, Chunyan Shi. Chenyang Wang. Effects of the “double reduction” policy on the commercial tutoring sector in China. *International Journal of Educational Development*, 2024. Volume 105, March 2024, 102989.

18. Nalyvaiko A. Education in China: Challenges and Prospects of the 21st Century. URL: <https://surl.lu/opkjzy>

19. Shparyk, O. M. An issue of differentiated instruction under the individualisation of education tendency. Reflections of the American, Ukrainian and

Chinese scholars. Вісник Київського національного лінгвістичного університету. Серія: Педагогіка та психологія. Випуск 31.

20. Shriyam Gupta, Chengfang Liu, Shaoping Li, Fang Chang, Yaojiang Shi Association between ability tracking and student's academic and non-academic outcomes: Empirical evidence from junior high schools in rural China. International Journal of Educational Development, 2023. URL: <https://www.sciencedirect.com/science/article/abs/pii/S0738059323002031>. 2019. С. 139-147.