

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ  
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

ННК “Інститут прикладного системного аналізу”

(повна назва інституту/факультету)

Кафедра Системного проектування

(повна назва кафедри)

«До захисту допущено»

Завідувач кафедри

\_\_\_\_\_ А.І.Петренко  
(підпис) (ініціали, прізвище)

“ \_\_\_\_ ” \_\_\_\_\_ 2016 р.

**Дипломна робота**

першого (бакалаврського) рівня вищої освіти  
(першого (бакалаврського), другого (магістерського))

зі спеціальності 7.05010102, 8.05010102 Інформаційні технології проектування  
7.05010103, 8.05010103 Системне проектування  
(код та назва спеціальності)

на тему: Інтегрована система навчання і контролю знань на основі адаптивних технологій

Виконав (-ла): студент (-ка) 4 курсу, групи ДА-21  
(шифр групи)

\_\_\_\_\_ Дяченко Олександр Сергійович \_\_\_\_\_  
(прізвище, ім'я, по батькові) (підпис)

Керівник професор, д.т.н. Снитюк В. Є. \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, прізвище та ініціали) (підпис)

Консультант Економічна частина професор, д.е.н. Семенченко Н.В. \_\_\_\_\_  
(назва розділу) (посада, вчене звання, науковий ступінь, прізвище, ініціали) (підпис)

Рецензент \_\_\_\_\_  
(посада, науковий ступінь, вчене звання, науковий ступінь, прізвище та ініціали) (підпис)

Нормоконтроль \_\_\_\_\_ ст. викладач Бритов О.А.

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент \_\_\_\_\_  
(підпис)

Київ – 2016 року

**Національний технічний університет України  
«Київський політехнічний інститут»**

Факультет (інститут) ННК “Інститут прикладного системного аналізу”  
(повна назва)

Кафедра Системного проектування  
(повна назва)

Рівень вищої освіти Перший(Бакалаврський)  
(перший (бакалаврський), другий (магістерський) або спеціаліста)

Спеціальність 7.05010102, 8.05010102 Інформаційні технології проектування  
7.05010103, 8.05010103 Системне проектування  
(код і назва)

ЗАТВЕРДЖУЮ  
Завідувач кафедри  
А.І.Петренко  
(підпис) (ініціали, прізвище)  
«\_\_» \_\_\_\_\_ 2016 р.

**ЗАВДАННЯ**  
**на дипломний проект (роботу) студенту**  
Дяченко Олександр Сергійович  
(прізвище, ім'я, по батькові)

1. Тема проекту (роботи) Інтегрована система навчання і контролю знань на основі адаптивних технологій  
керівник проекту (роботи) Снитюк Віталій Євгенович,  
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

затверджені наказом по університету від «\_\_» \_\_\_\_\_ 2016 р. № \_\_\_\_\_

2. Строк подання студентом проекту (роботи) 08.06.2016

3. Вихідні дані до проекту (роботи) \_\_\_\_\_

1. Мови програмування – PHP.
2. Форма реалізації – web-сайт.

4. Зміст розрахунково-пояснювальної записки (перелік завдань, які потрібно розробити)

1. Аналіз існуючих адаптивних і інтегрованих систем навчання.

2. Дослідити існуючі технології реалізації інтегрованих систем
3. Розглянути основні фреймворки.
4. Зробити оцінку використаних та досліджених підходів.
5. Визначити переваги та недоліки створеного рішення.
6. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслеників, плакатів тощо)

1. Модель адаптивного контролю знань та її модифікація – плакат.

2. Інформаційно-структурна модель студента – плакат.

6. Консультанти розділів проекту (роботи)

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Економічна частина	Семенченко Н. В., професор		

7. Дата видачі завдання 01.02.2015

#### Календарний план

№ з/п	Назва етапів виконання дипломного проекту (роботи)	Строк виконання етапів проекту (роботи)	Примітка
1	Отримання завдання	01.02.2016	
2	Ознайомлення з технічною літературою	15.02.2016	
3	Збір інформації	28.02.2016	
4	Дослідження предметної області та існуючих рішень	10.03.2016	
5	Аналіз існуючих методів генерації тестових питань та завдань	15.03.2016	
6	Дослідження існуючих технологій	25.03.2016	
7	Розробка методу генерації тестових питань по онтології	25.04.2016	
8	Дослідження існуючих варіантів АТ	30.04.2016	
9	Оформлення дипломної роботи	31.05.2016	
10	Отримання допуску до захисту та подача роботи в ДЕК	10.06.2016	

Студент

\_\_\_\_\_ (підпис)

О. С. Дяченко  
(ініціали, прізвище)

Керівник проекту (роботи)

\_\_\_\_\_

В. Є. Снитюк

## АНОТАЦІЯ

бакалаврської дипломної роботи Дяченка Олександра Сергійовича  
на тему «Інтегрована система навчання і контролю знань на основі адаптивних  
технологій»

Дипломна робота присвячена дослідженню адаптивних технологій контролю знань, аналізу вже існуючих технологій та внесення пропозицій щодо створення нової та запровадження її реалізації в системах дистанційного навчання. Актуальність теми зумовлена тим, що в останні роки в Україні підвищена увага приділяється методикам дистанційного навчання, важливою складовою яких є контроль знань. Ретельне вивчення принципів, логіки, технологій, ефективності, валідності й адаптивності контролю знань є актуальною і важливою проблемою, яка потребує вирішення

В роботі проведено аналіз адаптивних технологій, варіантів адаптивного тестування, способів представлення бази знань та методів генерації тестових питань.

Цілю дипломної роботи є створення прототипу інтегрованої системи навчання і контролю знань.

Загальний обсяг роботи 88 сторінок, 18 ілюстрацій, 6 таблиць 39 посилань.

Ключові слова: контроль знань, модель студента, модель викладача, комп'ютерне адаптивне тестування, генерація питань, адаптація.

## АННОТАЦИЯ

бакалаврской дипломной работы Дяченко Александра Сергеевича  
на тему «Интегрированная система обучения и контроля знаний на основе  
адаптивных технологий»

Дипломная работа посвящена исследованию адаптивных технологий контроля знаний, анализа уже существующих технологий и внесения предложений по созданию новой и внедрение ее реализации в системах дистанционного обучения. Актуальность темы обусловлена тем, что в последние годы в Украине особое внимание уделяется методикам дистанционного обучения, важной составляющей которых является контроль знаний. Тщательное изучение принципов, логики, технологий, эффективности, валидности и адаптивности контроля знаний является актуальной и важной проблемой, требующей решения

В работе проведен анализ адаптивных технологий, вариантов адаптивного тестирования, способов представления базы знаний и методов генерации тестовых вопросов.

Целью дипломной работы является создание прототипа интегрированной системы обучения и контроля знаний.

Общий объем работы 88 страниц, 18 иллюстраций, 6 таблиц 39 ссылок.

Ключевые слова: контроль знаний, модель студента, модель преподавателя, компьютерное адаптивное тестирование, генерация вопросов, адаптация.

## ABSTRACT

to the bachelor thesis by Diachenko Oleksandr Serhiyovych on “Integrated training system and knowledge supervision based on adaptive technology”

This thesis is devoted to research of adaptive control technology knowledge, analysis of existing technologies and making proposals for a new introduction and implementation of distance learning systems. Relevance of the topic due to the fact that in recent years in Ukraine increased emphasis on distance learning techniques, an important component of which is the control of knowledge. Careful study of the principles of logic, technology, efficiency, validity and adaptability of the control of knowledge is relevant and important problem that needs solving

The paper analyzes the adaptive technology options for adaptive testing, ways of presenting knowledge base and methods for generating test questions.

Aim of the thesis is to create a prototype of an integrated system of training and knowledge control.

The total volume of 88 pages, 24 illustrations, 6 tables 29 references.

Key words: control of knowledge, a model student, model teacher, computer adaptive testing, the generation of adaptation

## ЗМІСТ

<b>ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ.....</b>	<b>9</b>
<b>ВСТУП.....</b>	<b>10</b>
<b>1. АДАПТИВНІ ТЕХНОЛОГІЇ КОНТРОЛЮ ЗНАНЬ.....</b>	<b>12</b>
1.1 Еволюція контролю знань .....	12
1.2 Аспекти комп'ютерного контролю знань .....	14
1.3 Класифікація методів проведення контролю знань .....	17
1.4 Модель адаптивного контролю знань .....	21
1.5 Моделі студента та викладача.....	26
1.6 Адаптивне тестування.....	35
1.7 Висновки .....	40
<b>2. ГЕНЕРАЦІЯ ТЕСТОВИХ ПИТАНЬ ТА АЛГОРИТМИ ВИБОРУ НАВЧАЛЬНОГО МАТЕРІАЛУ .....</b>	<b>42</b>
2.1 Теоретичні відомості .....	42
2.1.1 Типи тестових питань .....	43
2.1.2 Методи генерації текстових завдань.....	49
2.2.1 Параметризовані тести .....	51
2.2.2 Семантичні мережі .....	54
2.2.3 Генерація тестових завдань на основі понятійно-тезисної моделі.	54
2.2.4 Формування тестових завдань на основі модифікації понятійно-тезисної моделі за допомогою ключових слів .....	55
2.2.5 Формування тестових завдань на основі модифікації понятійної тезисної моделі з системою семантичних класів .....	56
2.3 Метод генерації тестових завдань .....	57
2.4 Генерація тестування .....	63
2.5 Алгоритм вибору навчального матеріалу .....	70
2.6 Висновки .....	73
<b>3. Програмна реалізація, її тестування та результати.....</b>	<b>74</b>

3.1 Аналіз роботи програмного продукту.....	74
3.2 Висновки .....	77
<b>4. ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ .....</b>	<b>79</b>
4.1 ВСТУП.....	79
4.2 ПОСТАНОВКА ЗАДАЧІ .....	80
4.2.1 ОБҐРУНТУВАННЯ ФУНКЦІЙ ПРОГРАМНОГО ПРОДУКТУ ..	81
4.2.2 ВАРІАНТИ РЕАЛІЗАЦІЇ ОСНОВНИХ ФУНКЦІЙ.....	81
4.3 ОБҐРУНТУВАННЯ СИСТЕМИ ПАРАМЕТРІВ ПП .....	83
4.3.1 ОПИС ПАРАМЕТРІВ .....	83
4.3.2 КІЛЬКІСНА ОЦІНКА ПАРАМЕТРІВ .....	84
4.3.3 АНАЛІЗ ЕКСПЕРТНОГО ОЦІНЮВАННЯ ПАРАМЕТРІВ .....	85
4.4 АНАЛІЗ РІВНЯ ЯКОСТІ ВАРІАНТІВ РЕАЛІЗАЦІЇ ФУНКЦІЙ.....	89
4.5 ЕКОНОМІЧНИЙ АНАЛІЗ ВАРІАНТІВ РОЗРОБКИ ПП.....	90
4.6 ВИБІР КРАЩОГО ВАРІАНТА ПП ТЕХНІКО-ЕКОНОМІЧНОГО РІВНЯ .....	94
4.7 ВИСНОВОК .....	95
<b>ВИСНОВКИ .....</b>	<b>96</b>
<b>ПЕРЕЛІК ПОСИЛАНЬ.....</b>	<b>98</b>



## ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, СКОРОЧЕНЬ І ТЕРМІНІВ

КЗ	контроль знань	
ЕС	експертна система	
АСНКЗ	автоматизована система навчання і контролю знань	
КНС	комп'ютерна навчальна система	
АС	автоматизована система	
ООП	об'єктно-орієнтоване програмування	
ПТМ	понятійна-тезисна модель	
БЗ	база знань	
БД	база даних	
КАТ	комп'ютерне адаптивне тестування	
СДН	система дистанційного навчання	
МП	модульне проектування	
СП	системне	проектування

## ВСТУП

Останнім часом в усьому світі в системах освіти відбулися істотні структурні зміни, зумовлені розвитком Інтернет та його зростаючим впливом на всі сторони діяльності суспільства. Основну роль в удосконаленні системи освіти, без сумніву, мають нові інформаційні технології і, в першу чергу, дистанційні засоби навчання. На сьогодні у світі накопичено значний досвід реалізації систем дистанційного навчання.

Але системи дистанційного навчання ще не знайшли достатнього поширення в Україні. Тому в останні роки підвищена увага приділяється методикам дистанційного навчання, важливою складовою яких є контроль знань. Комп'ютерні системи контролю знань достатньо ефективні і дозволяють не тільки забезпечити державну перевірку якості знань, але й забезпечити основу самовдосконалення. Тому ретельне вивчення принципів, логіки, технологій, ефективності, валідності й адаптивності контролю знань є актуальною і важливою проблемою, яка потребує вирішення.

Все більше уваги приділяється контролю знань за допомогою тестування.

В сучасних навчальних системах тест повинен бути індивідуалізований. Кожен викладач розуміє, що добре підготовленому студенту немає необхідності давати легкі завдання, оскільки такі матеріали не володіють помітним потенціалом розвитку. Аналогічно, через високу імовірність неправильного рішення нема рації давати важкі завдання слабкому студенту, оскільки це сприяє зниженню навчальної мотивації. Виходом з цієї ситуації може стати адаптивне тестування – такий підхід до комп'ютерного тестування, який фактично дозволяє привести в стандартні групові тести елементи індивідуалізації, врахування індивідуальних особливостей даного випробуваного в процесі тестування.

Проблеми автоматизації процесу тестування і обробки його результатів достатньо повно досліджені в літературі. Однак, недостатній розвиток технологій генерації завдань істотно гальмує розвиток цього напрямку.

Питання формування самого банку завдань у більшості випадків залишається виключно прерогативою викладача, який працює без використання інтелектуальних засобів автоматизації даного процесу. Для вдосконалення адаптивних технологій контролю знань потрібно проаналізувати вже існуючі технології та виявити найкращу або запропонувати кращу. Також необхідно розробити метод генерації питань. Оскільки комп'ютерне адаптивне тестування є важливою і невід'ємною частиною контролю знань, необхідно проаналізувати вже існуючі варіанти та обрати найкращий.

Для вдосконалення адаптивних технологій контролю знань потрібно проаналізувати вже існуючі технології та виявити найкращу або запропонувати кращу. Також необхідно розробити метод генерації питань, що буде побудований. Оскільки комп'ютерне адаптивне тестування є важливою і невід'ємною частиною контролю знань, необхідно проаналізувати вже існуючі варіанти та обрати найкращий або запропонувати новий.

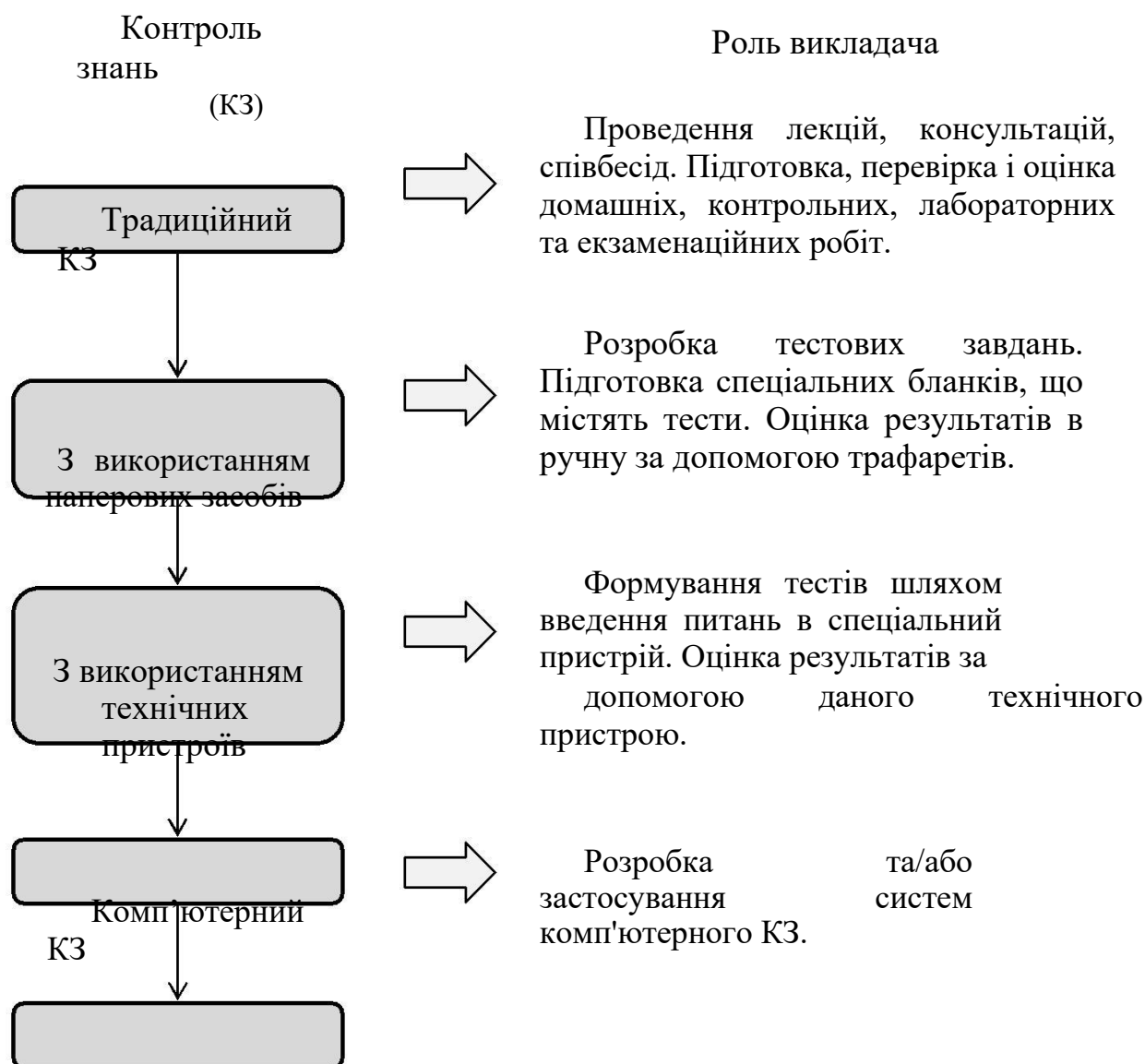
# 1. АДАПТИВНІ ТЕХНОЛОГІЇ КОНТРОЛЮ ЗНАНЬ

## 1.1 Еволюція контролю знань

З розвитком інформаційних телекомунікації та комп'ютерної техніки

відкриваються нові можливості в освітніх технологіях. Можна виділити п'ять

етапів еволюції розвитку контролю знань, які відображають форми його організації і роль викладача в цьому процесі (рис. 1.1).



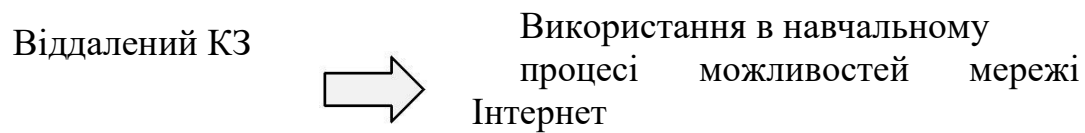


Рисунок 1.1 – Еволюція контролю знань[1]

Застосування в навчальному процесі того чи іншого підходу залежить від технічного та / або методичного забезпечення навчального закладу, а також від можливості використання викладачем у своїй роботі комп'ютерних технологій.

У порівнянні з традиційними формами КЗ, комп'ютерний контроль знань,

умінь і навичок має ряд переваг: використання новітніх методик перевірки і оцінки знань студентів, сучасних інформаційних технологій, можлива адаптація до індивідуальних характеристик студентів.

Однак, застосування комп'ютерних технологій у навчальному процесі потребує більш чіткого і однозначного визначення цілей контролю, відбору методичного матеріалу для оцінки знань і умінь студентів, з урахуванням мети проведеної перевірки, а також розробки моделі(ей) контролю та оцінки знань.[1]

## **1.2 Аспекти комп'ютерного контролю знань**

Проблеми комп'ютерного контролю знань зазвичай розглядаються у двох аспектах: методичному та технічному [2,3].

Методичні аспекти контролю знань пов'язані з вирішенням педагогічних і психологічних питань, тобто організація КЗ розглядається з точки зору дидактики. До методичним аспектам відносяться:

*Визначення типів і труднощі питань для перевірки знань, умінь і навичок студентів.* Завдання контролю – визначення відповідності підготовленості навчають того чи іншого рівня засвоєння навчального матеріалу. Оцінка якості знань на кожному рівні (знання, вміння, навички) може бути здійснена за допомогою використання різних типів завдань.

*Планування проведення контролю знань.* Навчальний процес прийнято розглядати як розподілений у часі процес формування необхідних знань,

навичок і вмінь. В даному випадку оцінювання відбувається поетапно і

дозволяє здійснити якісний і повний контроль. У залежності від часу проведеної перевірки розрізняють чотири види контролю знань: вихідний (попередній) контроль, поточний, рубіжний і підсумковий контроль.

*Визначення вимог до формування набору питань і завдань для опитування.* Це залежить від виду та мети контролю. Існують різні методи формування завдань для контролю: випадкова послідовність питань і завдань різної складності, труднощі й значимості; спеціальний набір завдань різної складності, сформований для перевірки певного або комплексного рівня підготовки і пред'являється в заданій послідовності та ін.

Технічний аспект пов'язаний, в першу чергу, з проблемою реалізації планованого контролю знань, з вибором відповідного алгоритму для оцінки контрольних робіт. До технічних аспектів відносяться:

*Формування набору контрольних завдань на основі обраного підходу.* З

урахуванням мети і виду проведеного контролю відбувається автоматична підготовка завдання (або набору завдань) для контролю і видача його студенту,

тобто управління контролем реалізується шляхом генерації контрольних завдань з урахуванням різних параметрів контролю знань.

*Вибір та використання в системі контролю параметрів КЗ.* Параметри контролю можна розділити на три групи: параметри, що характеризують окреме завдання і його виконання; параметри, що характеризують роботу учня з набором контрольних завдань; параметри, використовувані для настройки алгоритму (зазвичай задаються викладачем, але можуть мати і заздалегідь встановлені значення).

*Вибір алгоритму для оцінки знань студентів.* Будь алгоритм оцінки знань передбачає збір, аналіз та/або перетворення даних, одержуваних у процесі контролю, і формування самої оцінки (суми балів, рейтингу, рангу).

Розрізняють алгоритми, які застосовуються для виставлення оцінки тільки по завішені контролю, тобто на останньому етапі процесу оцінювання. Однак,

більшість алгоритмів використовуються паралельно з контролем знань, коли оцінка може бути виставлена за виконання окремого завдання, контрольної роботи або з дисципліни в цілому, при цьому отримана оцінка обов'язково



враховується у використовуваному методі проведення КЗ.[1]

### 1.3 Класифікація методів проведення контролю знань

Процес контролю знань складається з трьох етапів:

- 1) формування питань для КЗ на основі контрольних завдань, що зберігаються в БД;
- 2) видача їх студенту та отримання його відповіді, можливо, зі зворотним зв'язком;
- 3) виставлення оцінки.

Перші два етапи відносяться до організації процесу комп'ютерного контролю, на третьому етапі, на основі використовуваного алгоритму, обчислюється безпосередньо оцінка за контроль.

Таким чином, для управління контролем знань необхідна наявність:

- Методів і моделей організації (проведення) контролю;
- Моделей визначення та оцінки знань, умінь і навичок студента за результатами виконання контрольних завдань.

Методи проведення контролю знань можна розділити на три класи:

- Неадаптивні методи:
  - *сувора послідовність*; Набір завдань для контролю заздалегідь готується викладачем або розробником контрольної роботи і поміщається в БД системи. Як правило, це однакова послідовність питань для всіх студентів.

Недоліки цього методу очевидні: відсутність різноманітності (одна з вимог педагогіки), зниження самостійності виконання завдань та ін. Цей метод вважається найгіршим, тому й застосовується вкрай рідко. Метод можна дещо поліпшити, наприклад, підготувавши кілька варіантів контрольної роботи та/або видаючи завдання студентам у довільній послідовності.

- *випадкова вибірка*; Набір завдань формується безпосередньо перед контролем на основі завдань, що зберігаються в БД, тобто варіант

контрольної роботи - це  $n$  випадково вибраних завдань. Значення  $n$  може бути заздалегідь задано викладачем (розробником контрольної роботи) або вибрано студентом (наприклад, при самоперевірці). Перевага даного методу полягає в тому, що кожному студенту пропонується індивідуальна послідовність питань. Основний недолік методу - варіант контрольної роботи генерується без урахування складності завдань. Таким чином, набір завдань для одного студента може включати лише найважчі питання, а для іншого - тільки легкі. Це часто призводить до спотворення результатів контролю. Існують різні модифікації даного методу, що дозволяють враховувати метадані питань. Наприклад, а) можуть бути задані тема і загальний час контролю, час відповіді на кожне питання, число спроб дати відповідь і т. п.; б) додатково до (а) встановлюється число питань різного ступеня складності та/або з різних тем в кожному варіанті контрольної роботи.

- *Комбінований метод*; "Випадкова вибірка", доповнена "Суворою послідовністю". У цьому випадку викладач (розробник контрольної роботи) задає один або декілька питань, які неодмінно повинні бути включені в кожен варіант контрольної роботи. Інші завдання генеруються випадковим чином, як у другому методі.
- *Частково адаптивні методи*:
  - *випадкова вибірка з урахуванням окремих параметрів моделі студента (МС)*; Метод є розвитком не адаптивних методів КЗ. Він аналогічний "випадкової вибірки" та/або "комбінований метод", тобто набір завдань також формується безпосередньо перед контролем, але при генерації використовуються такі параметри МС, як загальний рівень підготовленості, здатність до навчання і, можливо, інші. Таким чином, кожному студенту генерується набір завдань, що відповідає його рівню підготовленості і здібностям, що є головною перевагою даного методу. Інша перевага методу: студент, виконуючи завдання, відповідні його

здібностям, не відчуває зайвої психологічного навантаження під час контролю. Як недолік даного методу можна відзначити наступне: студенти отримують завдання різної складності (це, безумовно, має бути враховано при виставленні оцінки), тобто один виконує тільки прості завдання, а інший - важкі. Тому, генеруючи питання студенту, відповідні його здібностям, доцільно включити в набір і один - два завдання підвищеної складності й значимості.

- *контроль на основі відповідей студента;* У цьому методі контроль здійснюється за заздалегідь складеним сценарієм або, іншими словами, за розгалуженою контролюючою програмою. Попередня підготовка сценарію КЗ дає можливість включити в програму питання різного ступеня складності і значущості, розташувавши найбільш значимі і важкі завдання в основній гілці програми, а більш прості - в розгалуженнях. Таким чином, студенти отримують різне число питань, а, отже, і час, що витрачається ними на контроль, різне, що є гідністю даного методу. Інша перевага методу - простота забезпечення зворотного зв'язку (видачі відповідного коментаря). Такий підхід, як один з методів проведення КЗ, був використаний в АОС "КОНТАКТ", в даний час зустрічається значно рідше, тому має істотний недолік: всім студентам пропонуються одні й ті ж завдання, одного разу включені в контролюючу програму. Усунути цей недолік досить просто - досить відокремити сценарій КЗ від набору контрольних завдань. Для цього необхідно підготувати комплект однотипних питань для кожного  $V_i$ , включеного в сценарій контролю, тобто  $V_i = \{V_{i1}, V_{i2}, \dots, V_{ik}\}$ , а в процесі контролю випадковим чином генерувати студенту питання з комплекту  $V_i$ .
- *контроль на основі моделі навчального матеріалу (НМ);* - У даному методі формування набору завдань для КЗ відбувається на основі моделі навчального матеріалу (курсу, теми, розділу теми), яка являє собою орієнтований граф: безліч вершин графа відповідає об'єктам вивчення, а

безліч ребер - зв'язкам між ними. Вивчення НМ, так само як і організація контролю, здійснюється відповідно до оптимальної послідовністю викладу навчального матеріалу, яка зазвичай є ніщо інше, як лінійна послідовність об'єктів вивчення. Таким чином, спочатку генерується завдання для перевірки знань першого навчального об'єкта, потім - другого і т. д., тобто послідовність видачі завдань аналогічна послідовності вивчення навчального матеріалу за моделлю НМ. При цьому, якщо планується перевірити і знання, і вміння, то одному навчальному об'єктові можуть відповідати кілька запитань. Такий підхід використовується в системі "Експерт-ТС", в якій модель НМ представлена у вигляді семантичної мережі.

- *модульно-рейтинговий метод*; Цей метод багато в чому аналогічний попередньому. Навчальний матеріал поділяється на окремі складові - модулі, для кожного з яких заздалегідь готується комплект контрольних завдань. У процесі КЗ студенту спочатку пропонується питання з першого модуля. При цьому після кожної відповіді студента обчислюється його рейтинг. Перехід до питань наступного модуля здійснюється при досягненні певного, заздалегідь встановленого рейтингу, причому студент з метою підвищення свого рейтингу, а, отже, і оцінки, може продовжити виконання завдань поточного модуля і лише потім перейти до наступного.
- Повністю адаптивні методи:
  - *контроль за моделлю студента*; У цьому методі враховуються багато параметрів моделі студента, а саме:
    - рівень підготовленості впливає на важкість запропонованих завдань;
    - вид репрезентативної системи обумовлює форму подання завдань (текст, візуальне зображення, використання звуку);
    - спрямованість особистості впливає на формулювання тексту

видається завдання;

- рівень занепокоєння-тривоги визначає як наявність зворотного зв'язку, так і форму, і детальність коментарів;
- особливості пам'яті є умовою для визначення часу виконання завдання та контрольної роботи в цілому;
- відповідь студента, точніше, правильність відповіді впливає на вибір наступного контрольного завдання.

Сценарій контролю зазвичай формується динамічно в процесі КЗ, хоча набір сценаріїв для різних груп студентів може бути створений і заздалегідь аналогічно методу "Контроль за відповідями студента".

- *контроль за моделями студента та навчального матеріалу*; Даний метод є розвитком попереднього, тобто при формуванні контрольних завдань використовуються наведені раніше параметри моделі студента, але процес КЗ будується на базі моделі навчального матеріалу, враховуючи взаємозв'язки між поняттями, що перевіряються.

## 1.4 Модель адаптивного контролю знань

Професором Л.А. Растрігіна [4] було запропоновано розглядати процес навчання як процес управління складною системою. Аналогічно можна представити і процес управління адаптивним контролем знань (рис. 1.2).



Рисунок 1.2 – Модель адаптивного контролю знаний  
[4]

Блок "*Алгоритм контролю*" виконує наступні функції:

- аналіз діяльності студента (перевірка правильності його відповідей і виконуваних дій);
- управління процесом контролю знань на основі обраного методу;
- визначення результатів контролю, яке зазвичай зводиться до виставлення оцінки студенту.

*База знань* (БЗ) містить методи і/або моделі процесу контролю, а також сукупність знань предметної області. *База даних* (БД) включає набори питань і завдань, призначених для перевірки знань студента та/або дані для формування завдань. Контрольні завдання можуть також генеруватися автоматично на основі БЗ. База даних і база знань спільно з моделлю студента утворюють репозиторій системи контролю.

*Модель студента* включає різноманітну інформацію про студента:

передісторія навчання; результати поточної роботи (тип виконаних завдань, час виконання завдань, число звернень за допомогою і т.д.); особистісні психологічні характеристики (тип і спрямованість особистості, репрезентативна система, здатність до навчання, рівень занепокоєння-тривоги, особливості пам'яті та ін.); загальний рівень підготовленості та інші. Модель студента динамічна, тобто змінюється в процесі проходження курсу, в ході роботи з системою.

*Генератор питань і задач* використовується для формування та видачі студенту чергового завдання (питання або завдання). Контроль знань здійснюється наступним чином: студент виконує запропоноване завдання, і результат його роботи поміщається в модель студента. Блок "*Алгоритм контролю*" на основі аналізу відповіді студента, цілей контролю  $Z$  і використовуваного методу проведення контролю, враховуючи зовнішні ресурси  $R1$  (наприклад, можливості системи контролю) і внутрішні ресурси студента  $R2$  (наприклад, час контролю), а також стан середовища  $Dx$ , визначає параметри завдання, яке має бути запропоновано студенту. Формувальник питань і завдань, отримавши від "*Алгоритму контролю*" дані про параметри наступного завдання, вибирає з БД та/або БЗ необхідну інформацію  $y$ , формує

текст завдання і видає його студенту. У найпростішому випадку робота цього блоку зводиться до вибору потрібного питання або завдання з бази даних. При деяких видах контролю (наприклад, при поточному КЗ або самоперевірки) може бути передбачений зворотний зв'язок К, який полягає у видачі коментаря на відповідь студента.

Таким чином, для управління адаптивним контролем знань необхідна наявність:

- методів і моделей організації (проведення) контролю;
- моделей визначення та оцінки знань, умінь і навичок студента за результатами виконання контрольних завдань. [5]

Але підготовка сучасного покоління фахівців з вищою освітою являє собою цілісний і вельми складний педагогічний процес, який реалізує багатоаспектний розвиток студентів. Не можна забувати, що, крім самостійного навчання, на знання студента, форму засвоєння матеріалу, його розуміння, а також ставлення до предмету впливають викладачі. Безпосередньо це стосується контролю знань.

Досліджуючи адаптивні технології контролю знань, ми часто стикаємося з поняттям «особливий підхід до кожного студента». Але ж значно частіше у вищих навчальних закладах можна почути «особливий підхід до кожного викладача». Це пов'язано з тим, що кожен викладач має власний погляд на предмет, навчальний матеріал, підхід до викладання, способи засвоєння студентами предмету, форми перевірки знань тощо. Тому було вирішено запропонувати модифікувати адаптивний метод контролю знань за моделями студента та навчального матеріалу, додавши *модель викладача*. Така модель повинна включати в собі, наприклад, такі параметри викладача:

- представлення предмету викладання (знання підручників з предмету, особисте бачення, власний вклад, пріоритетність закордонних даних(підручники, наукові статті) тощо)
- стиль викладання (задиктовування конспекту, демонстрування презентацій, обговорення зі студентами і т.д.)



- форма сприйняття студентами навчального матеріалу («зазубрювання», можливість викладу інформації своїми словами тощо)

Модель адаптивного контролю знань дещо модифікується з появою моделі викладача та її параметрів, а особливо зауважуючи її вплив на інші блоки.

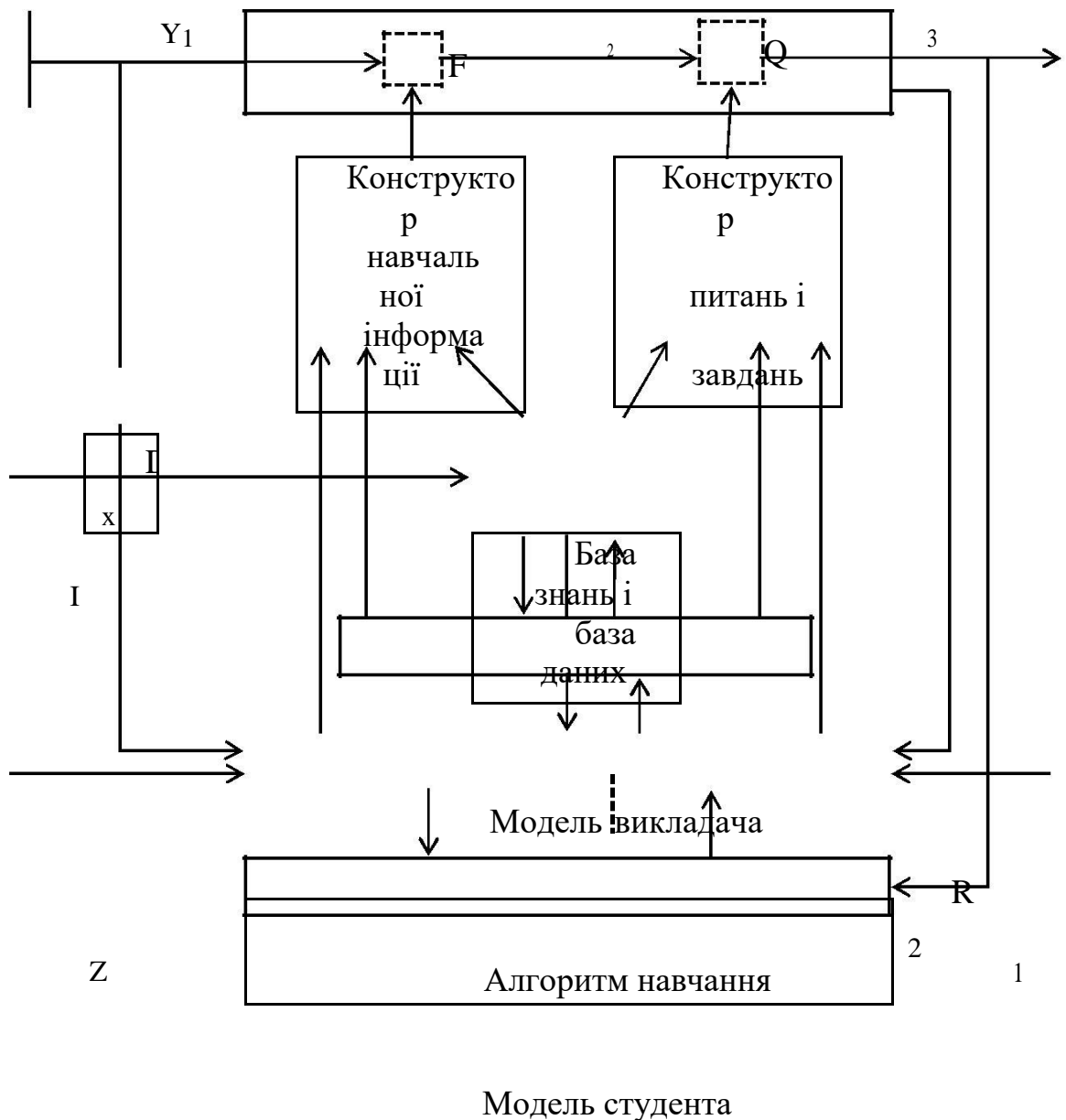


Рисунок 1.3 – Модифікована модель адаптивного контролю знань[13]

На початку навчання студент перебуває в стані  $Y_1$ . На основі інформації, що зберігається в базі знань і даних, а також за допомогою моделі викладача та алгоритму навчання формуються кадри навчального матеріалу. У ході вивчення наданої інформації учень отримує нові знання і переходить в стан  $Y_2$ . Для перевірки засвоєння матеріалу і формування умінь генеруються питання і завдання. Вони також формуються на основі бази знань і даних за допомогою моделі викладача та алгоритму навчання. Виконуючи надані питання та завдання, студент переходить у стан  $Y_3$ .

У базі даних зберігається загальна інформація про студентів, перелік навчальних курсів та ін. База знань включає загальновідому інформацію по кожному з предметів. С цього блоку під впливом алгоритму навчання та моделі викладача формуються два інших блоки:

- *конструктор навчальної інформації* – це ті знання (об'єм, вид, форма представлення тощо), що отримують студенті в процесі навчання.
- *конструктор питань і завдань* – необхідні дані для проведення контролю знань студентів.

*Алгоритм навчання* – це так званий «каркас» самого навчання. Він складається з 2 частин - модуля аналізу відповідей і модуля управління, який забезпечує управління процесом навчання з урахуванням інформації з бази знань і даних, мети навчання  $Z$ , ресурсів  $R$ , моделі студента і моделі викладача.

### **1.5 Моделі студента та викладача**

Модель студента є однією з базових компонент інтелектуальних комп'ютерних систем навчання. Вона містить досить повну інформацію про студента: рівень його знань, здатність до навчання, особистісні характеристики та інші параметри. Модель студента динамічна, тобто змінюється в процесі проходження курсу, в ході роботи з системою. [6]

Можна виділити три основні етапи процесу становлення знань про студента:

- *«Який він є»* (поведінкова модель).  
Реалізується на основі задачі діагностики знань та вмінь студента.
- *«Яким його хочуть бачити»* (нормативна модель).  
Використовується для порівняння з поточною поведінковою моделлю студента (також містить вимоги до особистісних якостей майбутніх фахівців)
- *«Яким він може стати»* (модель компетенцій).  
Визначає набір видів діяльності, які має здійснювати студент в майбутньому (тобто здатність застосовувати знання, вміння та особистісні якості в професійній сфері)

Класифікація моделей студента за В.А. Петрушиним[7]:

- Фіксуюча
  - Оверлейна
    - Векторна
    - Мережева
  - Генетичний граф
- Імітаційна
  - Модифікаційна
    - Модель помилок
    - Модель фальшивих правил
  - Виконувана («выводимая»)
    - Модель обмежень.

Багато з КНС не використовують модель студента, що знижує якість навчального процесу і не дозволяє організувати адаптивне навчання. Більшість систем, які засновані на моделях, реалізовані на базі оверлейних-векторних і мережних (графи знань). Однак вони не відображають всю

необхідну інформацію, а, як правило, включають тільки рівень знань. Досить рідко враховуються психологічні характеристики студента або, якщо беруться до уваги, то лише одна або дві. Для розробки адаптивних комп'ютерних систем навчання найбільш зручним є використання змішаної структури моделі студента. На рисунку 1.4 наведена інформаційно-структурна модель студента.

Особа викладача його ідейно-науковий, психолого-педагогічний та методичний рівні є вирішальним чинником у підготовці висококваліфікованих спеціалістів.

Відповідно до нової філософії освіти викладач вищої школи в сучасних соціокультурних умовах вбачається не просто транслятором науково - культурного й професійного досвіду, а носієм незаперечної істини, яка має бути засвоєна студентом. Центральний тягар в діалогічній культурі припадає на індивідуальність і індивідуальну свідомість, визнанні права на власну думку та позицію іншого, незалежно від соціально-рольової позиції, яку він обіймає.

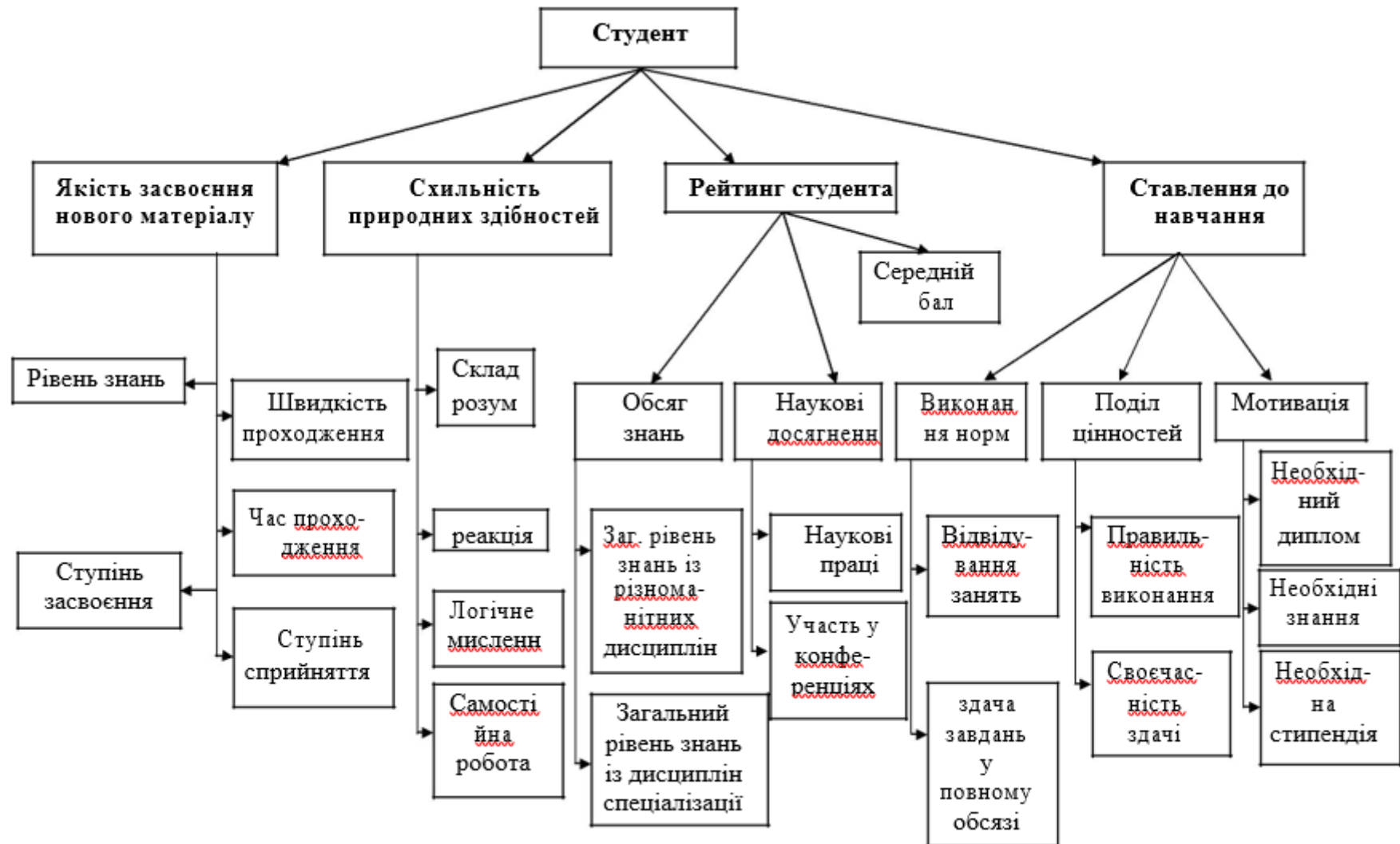


Рисунок 1.4 – Інформаційно-структурна модель студента[25]

Професійна діяльність викладача вищої школи вимагає наявності певних особистісних якостей, соціально-психологічних рис і педагогічних здібностей.

Серед них основні такі:

1. *Загальногромадянські риси:* широкий світогляд, принциповість і стійкість переконань, громадянська активність і цілеспрямованість, національна самосвідомість, патріотизм і толерантність щодо інших народів і культур, гуманізм і соціальний оптимізм, високий рівень відповідальності та працелюбність
2. *Морально-психологічні якості:* Чесність і ясність у взаєминах з людьми, високий рівень загальної психологічної культури, повага до професіоналізму інших і наукової спадщини, акуратність і охайність, дисциплінованість і вимогливість.
3. *Науково-педагогічні якості:* Науково-педагогічна творчість, професійна працездатність, активна інтелектуальна діяльність, науковий пошук, педагогічне спрямування наукової ерудиції, педагогічна спостережливість, педагогічна уява та інтуїція, володіння педагогічною технікою, активна інтелектуальна діяльність, науковий пошук, гнучкість і швидкість мислення у педагогічних ситуаціях, висока культура мови та мовлення, володіння мімікою, тоном голосу, поставою, рухами і жестами.
4. *Індивідуально-психологічні особливості:* високий рівень соціального

сприйняття й самопізнання ,висока інтелектуально-пізнавальна

зацікавленість і допитливість. інтерес до розвитку потенційних можливостей студентів і потреба в педагогічній діяльності з ними,

позитивна Я-концепція. високий рівень домагань, емоційна стійкість,

витримка й самовладання, саморегуляція, самостійність і діловитість у вирішенні життєво-важливих завдань, твердість характеру.

5. *Професійно-педагогічні здібності:* адекватне сприйняття студента,

безумовне прийняття його як особистості, педагогічний оптимізм, проектування цілей навчання, прогнозування шляхів становлення майбутнього спеціаліста, конструювання методичних підходів і здатність передбачати можливі результати, організаторські та комунікативні

здібності, духовний вплив на академічну групу і особистість студента,

організація розвиваючої інтеракції.

Особливість діяльності викладача вищої школи полягає в тому, що вона є складно організованою і складається з декількох взаємозв'язаних видів, які мають спільні компоненти. Викладач вузу здійснює такі види діяльності як :

педагогічна, науково-дослідна, професійна, адміністративно-господарча,

управлінська, комерційна і суспільна. Однак провідну роль в діяльності викладача відіграє педагогічна діяльність. Російський психолог Кузьміна

виділяє 5 рівнів продуктивності педагогічної діяльності:

1. *Репродуктивний*, коли педагог вміє розповісти іншим те що знає сам ;
2. *Адаптивний* , при якому в здатний адаптувати свою доповідь до вікових та психологічних особливостей аудиторії;
3. *Локально-моделюючий знання студентів*, коли педагог володіє стратегією навчання, знаннями, уміннями і навиками по окремим розділам курсу, що дозволяють визначити педагогічну мету, поставити завдання , розробити алгоритм їх вирішення і використовувати педагогічні засоби включення студентів в навчально-пізнавальну діяльність.
4. *Системно-моделюючий знання студентів*, коли педагог володіє стратегією формування системи знань, умінь і навичок з дисципліни в цілому.
5. *Системно-моделюючий діяльність і поведінку студентів*, при якому педагог може перетворити свою дисципліну у засіб формування особистості студента, його потреби до самовиховання, самонавчання і саморозвитку.

Аналіз літератури дозволяє виділити загальні вимоги до викладача вищої школи. Це перш за все наявність:

1. *професійної компетентності*, яка базується на спеціальній науковій, практичній і психолого-педагогічній підготовці;
2. *загальнокультурної і гуманітарної компетентності*, що включає в себе знання основ світової культури, гуманістичні особистісні якості, відповідальність за результати власної діяльності, мотивації до самовдосконалення;
3. *креативності*, що передбачає сформованість нестандартного мислення, володіння інноваційною стратегією і тактикою, пластичною



адаптацією до змін змісту і умов професійної діяльності;

4. *комунікативної компетентності*, що включає в себе розвинуту мову,

володіння іноземними мовами, сучасними засобами зв'язку і основами комп'ютерної грамотності;

5. *соціально-економічної компетентності*, що передбачає володіння основами сучасної ринкової економіки, знання законів бізнесу, азів екології і права.

Студент має сприйматись викладачем як суб'єкт навчання, їх взаємодія має будуватись на основі діалогічного підходу, що забезпечує суб'єкт-суб'єктні стосунки, які ґрунтуються на рівності позицій, повазі та довірі до студента як свого партнера . Саме це дає змогу зрозуміти один-одного і є найкращим способом взаємодії, а також допомагає задовольнити особисті потреби і інтереси всіх учасників навчального процесу.[8]

Ці дані, що входять до моделі викладача, створюють сам портрет викладача. Інші пов'язують цей блок з іншими в модифікованій моделі контролю знань. До них входять:

- Викладання

Стиль

- *Надиктовування конспекту*

Вид занять, під час якого студенти записують інформацію під диктовку викладача, або занотовують деякі її аспекти під час викладу на лекціях, семінарах та практичних.

- *Демонстрування презентацій*

Для покращення сприйняття інформації застосовують комп'ютерні презентації. Це набір слайдів, пов'язаних певного

тематикою. На них можна розміщувати текстові, графічні, відео- та звукові об'єкти.

- *Дискусія зі студентами*

Це метод навчання, який базується на обміні думками з певної проблеми. Точка зору, яку виражає студент у процесі дискусії, може як відображати його власну думку, так і спиратися на думки інших осіб.

- Бачення предмету

- ◆ Особисте бачення
- ◆ Власні розробки
- ◆ Підручники
- ◆ Закордонні відомості

- Форма засвоєння студентами матеріалу

*Зазубрювання*

Дослівне і в повному обсязі викладення учбового матеріалу під час контролю знань.

*Своїми словами*

Виклад учбового матеріалу своїми словами, тобто так як студент розуміє.

- Пріоритетність знань студентів по дисципліні

➤ Теорія

➤ Практика

➤ 50 на 50

- Вид занять

➤ *Лекції*

Основна форма проведення навчальних занять, призначених для

засвоєння теоретичного матеріалу. Усний виклад предмета викладачем, а

також публічне читання на яку-небудь тему.

- *Семінари*

Форма групових занять з предмета або теми, що відбувається під керівництвом викладача.

- *Практичні*

Форма навчального заняття, при якій викладач організує детальний розгляд студентами окремих теоретичних положень навчальної дисципліни та формує вміння і навички їх практичного застосування шляхом індивідуального виконання студентом відповідно сформульованих завдань.

- *Лабораторні*

Форма навчального заняття, при якому студент під керівництвом викладача, особисто проводить натурні або імітаційні експерименти, чи досліди з метою практичного підтвердження окремих теоретичних положень даної навчальної дисципліни, набуває практичних навичок роботи з лабораторним устаткуванням, обладнанням, обчислювальною технікою, вимірювальною апаратурою, методикою експериментальних досліджень у конкретній предметній галузі.

На рисунку 1.5 наведена інформаційно-структурна модель викладача.

## **1.6 Адаптивне тестування**

Адаптивне тестування – це широкий клас методик тестування, що передбачають зміну послідовності подання завдань в самому процесі тестування з урахуванням відповідей випробуваного на попередні завдання.

Відомо, що із значним збільшенням кількості завдань знижується ефективність самого тесту, так як випробовувані до кінця виконання тесту втомлюються і можуть неправильно виконати те завдання, яке виконали б правильно, якщо б воно знаходилося на початку тесту. Так, і зменшення кількості завдань тесту також не покращить його якість, оскільки не завжди малою кількістю запитань тесту можна охопити весь матеріал, який необхідно перевірити.

Використання завдань, що відповідають рівню підготовленості, істотно підвищує точність вимірювання і мінімізує час індивідуального тестування. У

свою чергу, зменшення кількості завдань у тесті та зменшення часу тестування дозволяє знизити витрати на проведення тестування.

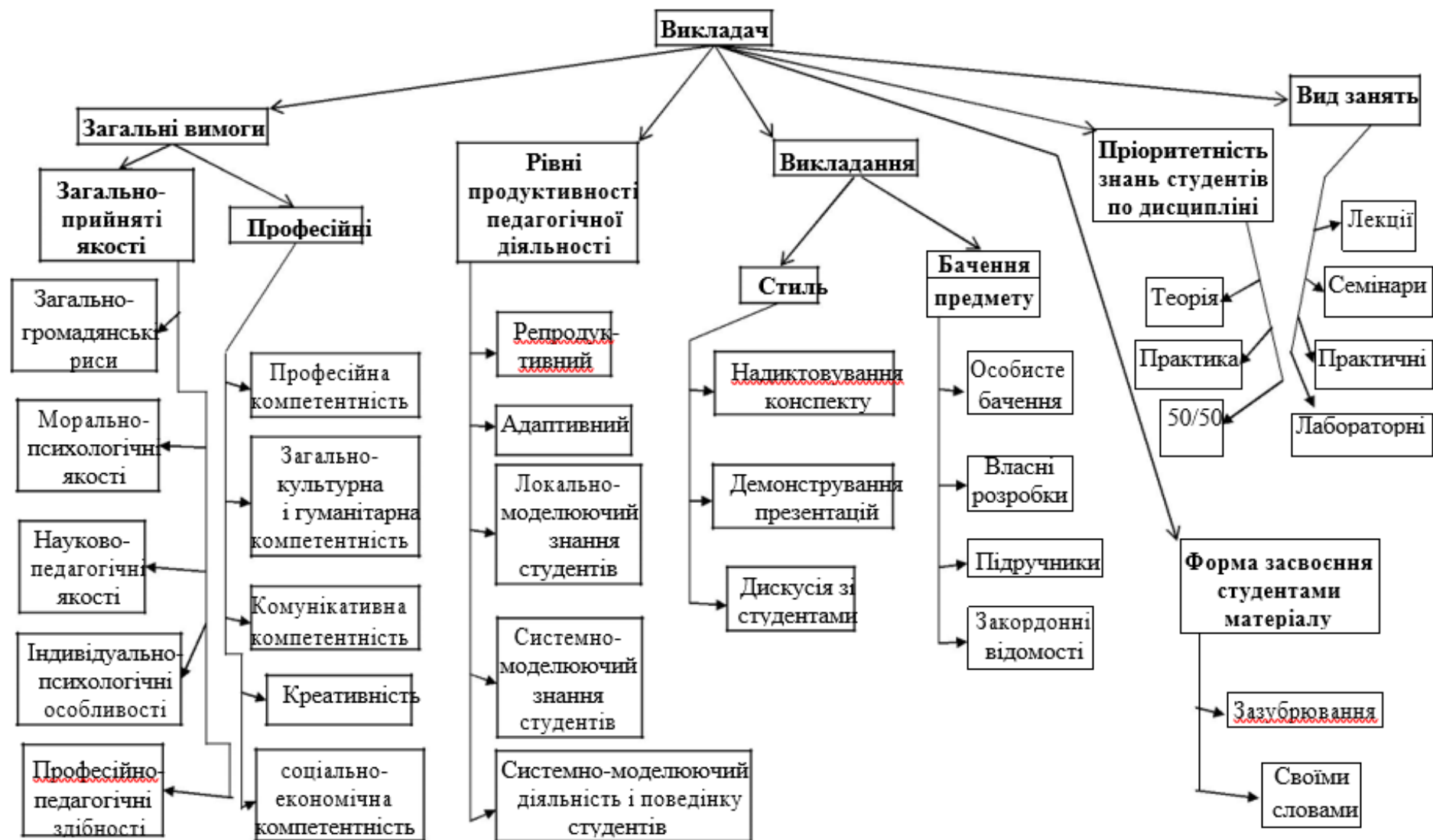


Рисунок 1.5 – Інформаційно-структурна модель викладача[25]



Використання завдань, що відповідають рівню підготовленості, істотно підвищує точність вимірювання і мінімізує час індивідуального тестування. У свою чергу, зменшення кількості завдань у тесті та зменшення часу тестування дозволяє знизити витрати на проведення тестування.

Види адаптивного тестування.

### 1. Двокрокове адаптивне тестування.

На першому кроці виявляється рівень підготовленості кожного випробуваного, на другому кроці відбувається саме тестування. Тест добирається для відповідного рівня підготовленості.

### 2. Групова адаптація.

Моделювання тесту відбувається з розрахунку на задану групу.

Проводиться вступне тестування та визначаються межі рівнів підготовленості студентів даної групи, далі будується сам тест з розрахунку на відповідний

(визначений на попередньому етапі) рівень підготовленості.

### 3. Багатокрокове тестування.

Багатокрокова стратегія адаптивного тестування проводиться тільки в комп'ютерному режимі і поділяється на фіксовано розгалужені і варіативно-розгалужені залежно від того, як конструюються багатокрокові адаптивні тести.[9]

У західній літературі виділяється три варіанти адаптивного тестування:

#### 1) Пірамідальне тестування

При відсутності попередніх оцінок на першому кроці всім випробовуваним видаються завдання однакового середнього рівня важкості, який визначається як середнє між найнижчим і найвищим рівнем. Якщо відповідь на питання неправильна, то важкість наступного питання буде визначатися як середнє між найнижчим рівнем важкості і поточним, на який він не відповів, а при правильній відповіді – між найвищим і поточним. Таким чином відбувається постійне ділення шкали складності завдань навпіл.;

## 2) *Flexilevel-контроль*

Контроль починається з будь-якого рівня складності, який обирає сам той, хто проходить тестування, а потім відбувається поступове наближення до реального рівня підготовленості;

## 3) *Stradaptive (від англ. stratified adaptive)*

Тестування проводиться за допомогою банку тестових завдань (БТЗ), де завдання розділені за рівнями складності. При правильній відповіді, наступне завдання береться з більш високого рівня складності, при неправильній - навпаки. Наступне завдання відрізняється від попереднього на один крок по складності [10,11]. Схематично зображено метод на рис 1.6

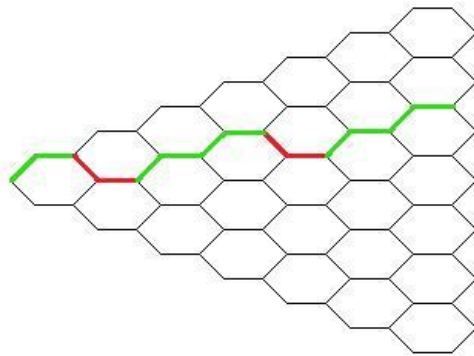


Рисунок 1.6 – *Stradaptive* тестування(правильна відповідь – зеленій шлях, неправильна - червоний)[10]

## 1.7 Висновки

В результаті досліджень було запропоновано ввести в адаптивну технологію контролю знань, крім моделі студента, ще й моделі викладача. Ця модель впливає майже на всі компоненти моделі контролю знань. Головна її задача поєднати в одне ціле навчальну інформацію, що дається викладачем на лекціях і питання та завдання, що формують перевірку знань студентів. Також запропоновано інформаційно-структурні моделі викладача та студента. Вони складаються з загальної кваліфікаційно-психологічна характеристики і з компонентів, що впливають на формування питань у випадку викладача і формування тесту у випадку студента.

Введення нової моделі змінило модель адаптивного контролю знань,



адже вплив першої поширюється майже на всі компоненти останньої. Такі зміни значно покращать уявлення про систему контролю знань успішності студентів.

## 2. ГЕНЕРАЦІЯ ТЕСТОВИХ ПИТАНЬ ТА АЛГОРИТМИ ВИБОРУ НАВЧАЛЬНОГО МАТЕРІАЛУ

### 2.1 Теоретичні відомості

Тест – сукупність запитань, які переважно вимагають однозначної відповіді, укладений за певними правилами та процедурами, передбачає попередню експериментальну перевірку й відповідає таким характеристикам ефективності, як валідність і надійність. Тестологія – міждисциплінарна наука про створення якісних та науково обґрунтованих вимірювальних діагностичних методик тестування. Тестування – метод психологічної діагностики, що використовує стандартизовані питання і завдання (тести), що мають певну шкалу значень. Застосовується для стандартизованого вимірювання індивідуальних відмінностей.

Існують три основні сфери застосування тестування:

- 1) освіта - у зв'язку із збільшенням тривалості навчання і ускладненням учбових програм;
- 2) професійна підготовка і відбір - у зв'язку із збільшенням темпу зростання і ускладненням виробництва;
- 3) психологічне консультування - у зв'язку з прискоренням соціодинамічних процесів. Тестування дозволяє з відомою вірогідністю визначити актуальний рівень розвитку у індивіда необхідних навиків, знань і особових характеристик.

Використання комп'ютерних програм для проведення такої форми контролю знань як тестування стає все більш поширеним явищем.

Автоматизовані системи контролю знань використовуються як окремо, так і як складові частини систем дистанційного навчання. На сучасному етапі в практиці педагогічного тестування намічений перехід від

найпростішого засобу оцінки знань по числу правильних відповідей до складніших методів,

заснованих на математичних моделях сучасної теорії параметризації і

моделювання педагогічних тестів.

### 2.1.1 Типи тестових питань

Дослідження функціональних можливостей найбільш поширених систем тестування та модулів контролю знань в комп'ютерних системах навчання показало, що можна виділити такі типи тестових завдань:

1) питання закритого типу, в якому користувачу надаються варіанти відповіді:

- a. Одиночний вибір – вибір одного варіанта відповіді з декількох запропонованих. Питання можна представити у вигляді конструкції  $P = \langle A, B, i, R \rangle$ , де  $A$  – текст питання,  $B$  – множина варіантів відповіді,  $i$  – номер вірної відповіді,  $R$  – посилання на теоретичний матеріал.
- b. Множинний вибір – вибір одного або декількох варіантів відповіді з декількох запропонованих. Питання можна представити у вигляді конструкції  $P = \langle A, \langle B_i, C_i \rangle, R \rangle$ , де  $A$  – текст питання, список відповідей складається з  $B_i$  – текст  $i$ -того варіанту відповіді і  $C_i$  – правильність  $i$ -того варіанту  $(0,1)$ ,  $R$  – посилання на теоретичний матеріал.
- c. Відповідність – упорядкування висловлювань у двох списках так, щоб вони відповідали один одному. Питання можна

представити у вигляді конструкції  $P = \langle A, \langle V_i, C_i \rangle, R \rangle$ , де  $A$  – текст питання, списки відповідей складаються з:  $V_i$  –  $i$ -тий елемент списку 1,  $C_i$  –  $i$ -тий елемент списку 2,  $R$  – посилання на теоретичний матеріал.

- d. Упорядкування списку – розстановка відповідей у певній послідовності. Питання можна представити у вигляді конструкції  $P = \langle A, \langle V_i \rangle, R \rangle$ , де  $A$  – текст питання,  $V$  – список відповідей,  $R$  – посилання на теоретичний матеріал.
- e. Встановлення істинності висловлювання: вибір одного з двох запропонованих варіантів відповіді «так» чи «ні». Можливий також вибір трьох градацій: «так», «ні», «не знаю» або п'ятих: «так», «скоріше так», «не знаю», «скоріше ні», «ні» – зазвичай використовуються в психологічних тестах. Питання аналогічне одиночному вибору.
- f. Обробка списків:
- I. перетягування: є набір елементів – «приймачів», які потрібно перемістити в певні області екрану – «джерела» в якості відповіді на питання;
  - II. переміщення та з'єднання: питання виглядає як набір «об'єктів» і «з'єднувачів», необхідно розмістити та з'єднати елементи особливим чином;
  - III. створення дерева: є два списки: зліва знаходиться дерево, справа – елементи, які потрібно розмістити в дереві;
  - IV. створення та впорядкування списку: з декількох списків елементів потрібно скласти один – перетягнути елементи і розмістити їх у правильній послідовності.

2) Питання закритого типу, в яких надання варіантів відповіді користувачу не передбачено:

- a. Уведення тексту, числа, дати або часу.
- b. Мовна відповідь: відповідь на питання за допомогою мікрофона.
- c. Завантаження файлу в якості відповіді.
- d. Графічна відповідь:
  - I. вибір певної області на зображенні в якості відповіді;
  - II. малювання точки / лінії / кола / відрізка / прямокутника в якості відповіді на запитання;
  - III. малювання відповіді з використанням примітивного графічного редактора.
- e. Уведення формули в якості відповіді за допомогою убудованого редактора формул. Такий редактор реалізований, наприклад у програмі KTest , коли формула записується і зберігається у вигляді рядка, а вигляд математичного виразу приймає синхронно введенню у спеціальному вікні. Для створення вбудованого редактора формул доцільно використовувати систему верстки TeX, розроблену Д. Кнудом. TeX – стандарт de facto в математичному співтоваристві й є продуктивною при великих обсягах роботи.

3) У комбінованій формі питання наявність варіантів відповіді на питання залежить від налаштувань. До такого типу належить заповнення пропусків у тексті питання.

Класифікація тестових питань наведена на рис. 2.1.

При відкритій формі питання відповідь вводиться користувачем

клавіатури. Передбачаються такі варіанти перевірки відповіді:

- оцінка відповіді користувача викладачем вручну після автоматичної перевірки інших відповідей системою.

Приклад:

*Яку геометричну фігуру називають квадратом? Квадрат – це чотирикутник, у якого всі сторони рівні – неповністю правильна відповідь.*

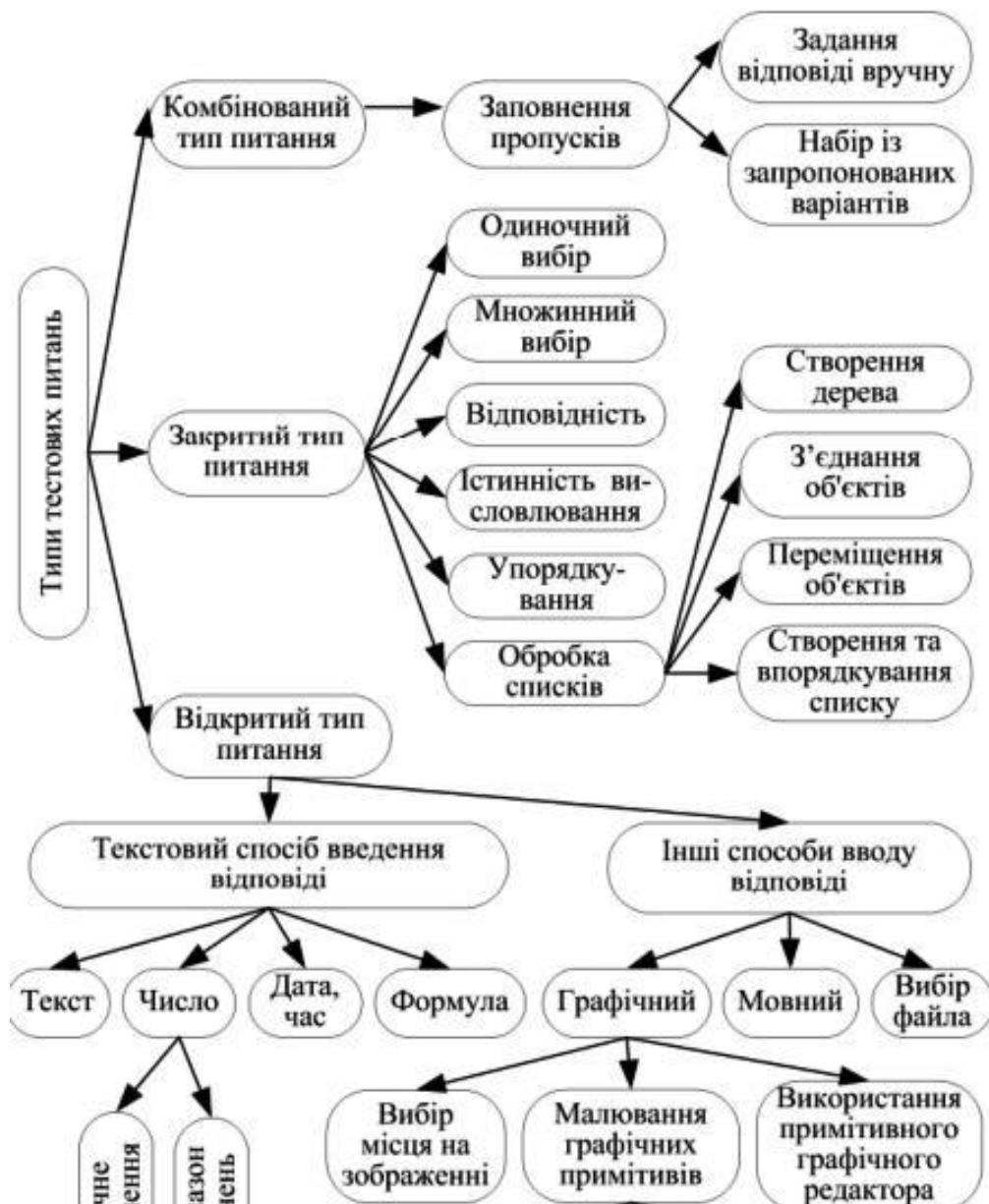


Рисунок 2.1 – Класифікація типів тестових питань  
[28]

- перерахування відповідей, які можна зарахувати як правильні – завдання буде вважатися виконаним вірно, якщо користувач вкаже один з них.

Приклад:

*Найвища гора світу: Джомолунгма / Еверест.*

- перерахування всіх правильних відповідей завдання буде вважатися виконаним вірно, якщо користувач вкаже всі відповіді в довільному порядку.

Приклад:

*Перерахуйте усі материки, які є зараз на планеті Земля: Австралія; Антарктида; Африка; Євразія; Південна Америка; Північна Америка.*

- перерахування всіх правильних відповідей у визначеному порядку – завдання буде вважатися виконаним вірно, якщо користувач вкаже всі відповіді у тому ж порядку.

Приклад:

*Назвіть місяці року англійською мовою. January, February, March, April,*

*May, June, July, August, September, October, November, December.*

- використання системи шаблонів на основі регулярних виразів для оцінки відповіді користувача.

Приклад:

*Найвища гора світу: Джомолунгма | Еверест.*



## 2.1.2 Методи генерації текстових завдань

У навчальних системах використовуються два основні підходи до організації контролю знань:

1. Оцінка дій студента застосовується в експертно-навчальних системах, тобто навчальних системах, що ґрунтуються на знаннях. Знання про предметну область і правила оцінки дій студента дозволяють системі визначати рівень знань в ході діалогу. За допомогою методу можна на високому рівні моделювати взаємодію викладача зі студентом.
2. Стандартизований контроль знань полягає в тому, що студенту пропонується вибірка спеціальних завдань і з відповідей на неї виноситься судження про його знання. Для вимірювання здібностей даним методом необхідно проводити періодичні перевірки. Стандартизовані методи контролю знань мають наступні позитивні властивості, що визначають доцільність їх застосування: короткочасність перевірки; стандартність проведення перевірки та аналізу результатів; можливість представлення результатів перевірки в числовій формі та їх математичної обробки.

Розглянемо докладно методологію генерації тестових завдань (рис. 2.2).

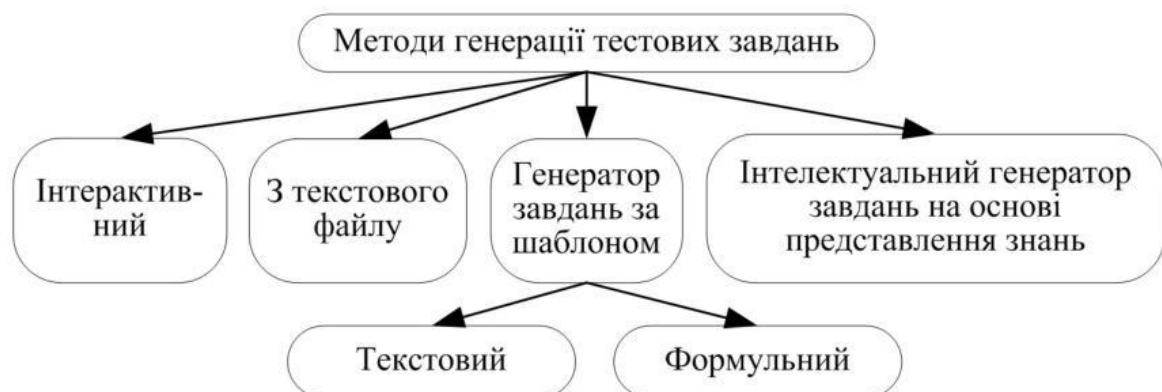


Рисунок 2.2 – Класифікація методів генерації завдань [28]

Базу завдань можна створювати у таких режимах:

**1. Інтерактивний.** Тестові завдання готуються заздалегідь і послідовно вводяться в базу за допомогою вбудованого в програму текстового редактора. Такий спосіб введення даних дає змогу виконувати довільне форматування завдань та використовувати мультимедійний контент.

Для реалізації навчального процесу питання необхідно згрупувати за темами з можливістю подальшого автоматичного вибору визначеної кількості питань з кожної теми при тестуванні. Такий поділ дає змогу перевіряти рівень знань і проводити аналіз оволодіння матеріалом з окремої теми. Крім того, групування питань за темами більш зручне для сприйняття.

**2. Імпорт тестових завдань,** записаних з використанням відповідного синтаксису, із текстового файлу. Такий спосіб уведення завдань дає змогу готувати завдання разом із відповідями за межами системи тестування, наприклад, за допомогою найпростішого текстового редактора «Блокнот». Тестові завдання генеруються шляхом довільної вибірки визначеної кількості питань з кожної теми та переміщення питань і відповідей у тесті. Такий механізм генерації виключає передачу інформації від одного учасника тестування до іншого про перелік питань та позиції правильних відповідей для можливого списування іншими учасниками і, таким чином, студент не може скористатися попереднім знанням тесту, із заздалегідь відомою послідовністю запитань і відповідей.

**3. Автоматизований.** Наявність такого режиму якісно відрізняє систему тестування від інших подібних програм. При використанні такого режиму автор тесту створює шаблон питання й варіанта відповіді, за яким програма тестування буде автоматично генерувати задану кількість завдань із заданою кількістю варіантів відповіді. Цей шаблон уводиться у систему за допомогою текстового редактора. За заданим шаблоном може генеруватися практично

необмежена кількість завдань.

4. Інтелектуальна генерація тестових питань на основі моделей представлення знань. [28]

### 2.2.1 Параметризовані тести

Метод параметризованих задач дозволяє генерувати завдання відкритого типу. Тестований при цьому, як правило, повинен ввести деяке число, яке і буде результатом рішення представленої задачі.

У методі параметризованих задач використовується принцип фасетності, що дозволяє створювати, в одному завданні, відразу кілька варіантів. Кожному студенту програма видає тільки один елемент з фасета. Розглянемо приклади фасетних завдань з одним фасетом і двома варіантами:

1. {Зимовий, літній} мусон дме

а) з суші на море

б) з моря на сушу

Параметризоване питання являє собою шаблон питання, створений автором. У момент видачі, шаблон доповнюється параметром, значення якого генерується в заздалегідь встановлених межах. Під шаблоном звичайно розуміють заготовку тексту, в якому деякі елементи можна змінювати відповідно до заданого алгоритму. Суть методу полягає в тому, що маючи шаблон завдання і змінюючи параметри на вході, ми отримуємо нові варіанти завдання на виході.

В якості прикладу авторами було наведено наступне завдання:

*У Петра було два яблука, а у Миколи три. Скільки яблук було у Петра і Миколи?*

Для того, щоб зробити з цього завдання шаблон, необхідно замість

конкретних чисел поставити параметри і алгоритми, що генерують значення цих параметрів. Тоді ця задача може бути записана як: У Петра було  $gen(x)$  яблука, а у Миколи  $gen(y)$ . Скільки яблук було у Петра і Миколи?

Тут  $gen(x)$  і  $gen(y)$  - програма, що генерує значення для змінної  $x$  і  $y$ , відповідно. До шаблону потрібно прикласти програму вирішення задачі по згенерованим параметрам. Тоді шаблон завдання буде виглядати наступним чином: правильна відповідь:  $(rez = solv(x,y))$ , де  $solv(x,y)$  - програма обчислення правильної відповіді.

Недоліком цього методу є трудомісткість формування набору шаблонів завдань. Перевага цього методу полягає в тому, що для малої кількості шаблонів можна згенерувати достатньо велику кількість завдань. Так для наведеного прикладу, якщо параметр  $x$  може прийняти 10 різних значень, а параметр  $y$  - 15, тоді ми отримаємо можливість згенерувати 150 варіантів завдань.

Генерація питань на основі алгоритмів - окремий випадок параметризованих задач. В основі питання навчають, пропонується деякий програмний код, який реалізує певний алгоритм. Тестований повинен визначити значення деякого параметра алгоритму, тим самим демонструючи своє розуміння мови програмування. Суть методу полягає в наступному: ґрунтуючись на умовах виходу з циклу, можна побудувати генератор питань.

Розглянемо приклад знаходження суми натурального ряду:

Крок 1:  $i = 0, S = 0,$

Крок 2:  $S = S + i, i = i + 2,$

Крок 3: якщо  $i < n$ , то перейти на крок 2.

Питання 1. Яке значення прийме змінна  $S$  після завершення циклу, якщо  $n = генерувати()$ .

Питання 2. Яке значення змінної  $n$  було встановлено, якщо по

завершенню циклу значення  $S = \text{генерувати} ()$ .

Питання 3. Скільки ітерацій було виконано, якщо по завершенню циклу значення  $S = \text{генерувати} ()$  і т.п.

Генерація тестових завдань на основі дерев та/або. Методів побудови алгоритмів генерації заснований на використанні дерев та/або, дозволяє представити будь-яку комбінаторну множину у вигляді дерева та/або. На рис. 2.3 представлено опис примітивного текстового завдання з математики.

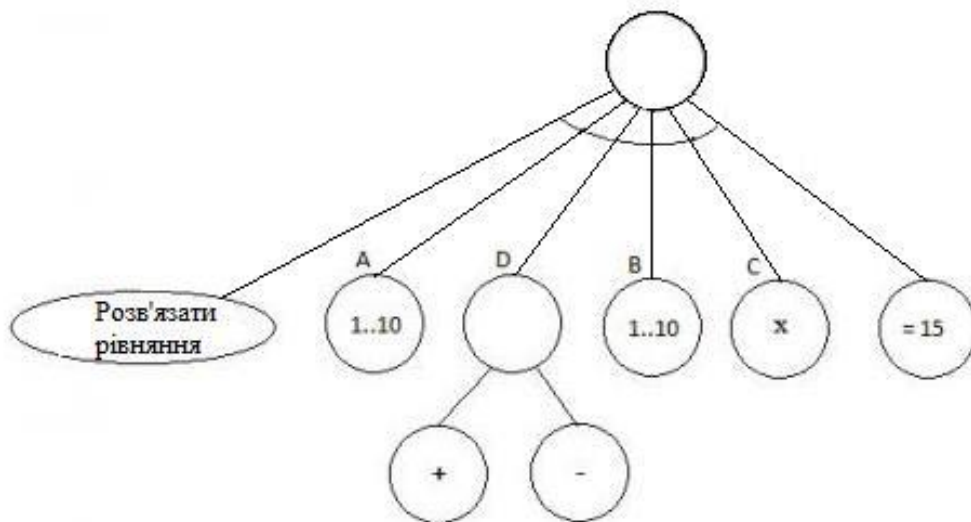


Рисунок 2.3 – Опис текстового завдання у вигляді дерева та/або [29]

Розглянемо гілку D. Гілка D має два або-варіанти представлення, перший варіант символ «+», а другий - «-». В даному випадку завдання буде мати 200

варіантів представлення. Однак, у випадку вибору в гілці D знака «+», рішення даного завдання буде формула  $C = (15-A) / B$ , при знаку «-» -  $C = (15-A) / -B$ .

Тобто рішення даної задачі також можна представити у вигляді дерева та/або

(рис. 2.4).

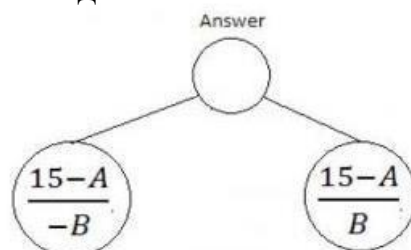


Рисунок 2.4 – Гілка дерева та/або з варіантами відповіді [29]  
Недолік методу полягає в складності формуванні шаблону при більш складному завданні. [29]

### **2.2.2 Семантичні мережі**

Суть застосування семантичних мереж полягає в генерації тестових питань на основі бази знань (БЗ), яка складається експертом. Структурною одиницею бази знань є тріада: «поняття» - «ставлення» - «поняття». Експерт з предметної області наповнює БЗ, а формування тестів відбувається автоматично шляхом опущення однієї з ланок тріади. Недоліки цього підходу

полягають великих трудових витратах на формування бази знань, необхідності залучення експерта з предметної області та інженера по знаннях.[29] Також можна виділити такі недоліки – відсутність формалізації алгоритму побудови завершеної цілісної семантичної мережі, яка б коректно відображала предметну область, а також лінгвістична незрозумілість, а іноді – недоцільність завдань, що генеруються.[30] Семантичні мережі як і інші моделі знань, що застосовуються для класичних задач штучного інтелекту, погано підходять для освітніх цілей зокрема для формування якісних тестових завдань.[29]

### **2.2.3 Генерація тестових завдань на основі понятійно-тезисної моделі**

ПТМ - це модель подання знань, яка формалізує зміст навчального матеріалу. Для представлення знань про поняття в моделі існують структурні елементи - відомості про об'єкт, або тези про поняття.

Семантичні елементи ПТМ виділяється безпосередньо з тексту навчального фрагменту. Тестове завдання формується шляхом вибору

контрольної понятійно-тезисної пари і дистракторів на основі інших понять чи тез. Перевагою підходу є простота формування БЗ і висока якість тестових завдань у порівнянні з іншими методами автоматичної побудови тестів. [29]

Недоліки методу:

- з педагогічної точки зору надмірне обмеження структури тесту рамками понять та тезисів, оскільки в практиці формування тестових завдань часто трапляються „тезисно-тезисні” конструкції доволі складної структури;
- неструктурованість представлення тез у даному методі обмежує складність тестового завдання, оскільки не аналізуються окремі частини самої тези;
- запропонована авторами структура класів не обґрунтована, що породжує невизначеність у їх наповненні й може спричинити формування альтернатив з низькою педагогічною цінністю. [30]

#### **2.2.4 Формування тестових завдань на основі модифікації понятійно-тезисної моделі за допомогою ключових слів**

В роботі [31] пропонується модифікувати ПТМ шляхом додавання до структурних елементів ще однієї складової - ключових слів.

Ключові слова - це слова або словосполучення, які використовуються для вираження деякого контекстного аспекту змісту тези про поняття. Вони несуть істотну смислове навантаження і так чи інакше характеризують поняття, про яке йдеться в тезі.

Під час семантичного розбору тексту викладач виділяє поняття, додає до них тези і виділяє ключові слова.

Наведемо приклад навчального матеріалу і виконаємо його семантичний

розбір, згідно удосконаленої ПТМ і її базових елементів. В якості навчального матеріалу був узятий фрагмент курсу лекцій «Програмування на мові Сі ++».

Виділяємо з нього наступні структурні елементи:

*Поняття:* Посилання

*Набір тез:*

- Є видозміненою формою покажчика;
- Може розглядатися як покажчик, який є ще одним ім'ям або псевдонімом змінної;
- Можна ініціалізувати тільки один раз;
- Підвищують ефективність програми, особливо при передачі даних при виклику функції.

*Ключові слова:* покажчик (2), змінна (1), програма (1), функція (1).

Аналізуючи ключові слова в тезах, доцільно внести в БЗ таке поняття як «Покажчик», адже воно зустрічається досить часто в тексті тез і дидактично передують поняттю «посилання», а також є ключовим на думку упорядника тесту.

Отже далі проробляємо такі ж дії вже з поняттям «Покажчик».

Даний підхід є перспективним з точки зору розширення варіантів тестових завдань. У той же час слід зазначити, що ПТМ передбачає автоматичний синтаксичний аналіз тексту тез на предмет входження в нього інших понять, що фактично замінює ручну працю. [29]

### **2.2.5 Формування тестових завдань на основі модифікації понятійної тезисної моделі з системою семантичних класів**

В цьому методі для повного опису характеристик предметної області на базі ПТМ пропонується сформувати такі когнітивні абстрактні класи: визначення, проблеми, методи, ефективність методів, приклади реалізації



методів.

Приклад системи семантичних класів, зображено на рис. 2.5.

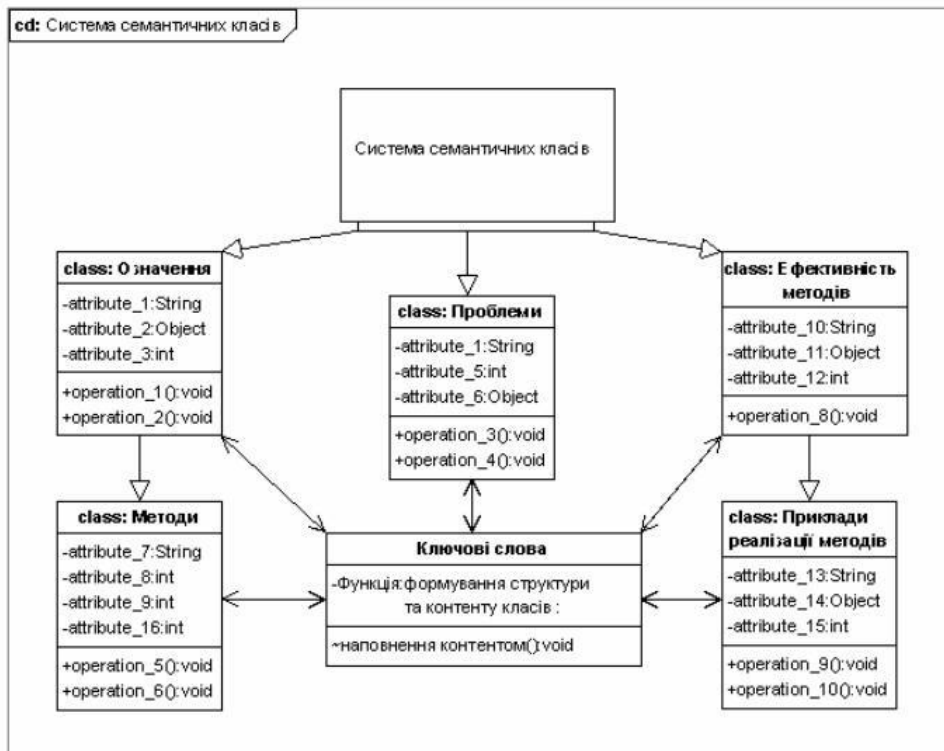


Рисунок 2.5 – Система семантичних класів [29]

Суть методу полягає в тому, що для генерації тесту деяке твердження розбивається на основну та альтернативну частини, які можуть містити одну або декілька компонент. Альтернативна частина тесту поповнюється аналогічними з лінгвістичного змістом, синтаксично узгодженими частинами інших тверджень. Синтаксичне узгодження забезпечується, коли зв'язку компонент тверджень однорідні за типами та числами.

На основі бази тверджень може бути сформований набір тестових завдань із заданими параметрами у вигляді динамічної структури.

Запропоновано вирішення мовної неузгодженості, характерної для деяких тестових завдань, що генеруються на основі ПТМ. У той же час, даний підхід передбачає значні трудові витрати на формалізацію додаткових характеристик і сутностей, що доповнюють базову версію ПТМ.[29]

## 2.3 Метод генерації тестових завдань

За своєю сутністю цей метод подібний до методу генерації тестових завдань на основі семантичних класів. Оскільки як зазначалося раніше в розділі 2, графічно онтологія має вигляд мережі, вершини якої є термінами і відносинами, а ребра вказують на зв'язки між ними. Зазначені відносини дозволяють безпосередньо формувати питання і генерувати відповіді. Якщо використовується такий тестування, то неправильні відповіді можуть генеруватися з відповідних складових інших концептів.

Враховуючи специфіку онтології навчального курсу і її подальше використання для контролю знань, визначаємо домінуючі відносини. Їх типи: «Атрибути», «Визначення», «Частина-Ціле», «Складова частина», «Дія», «Стан» та інші.

Концепти онтології з множини  $X$  між собою знаходяться в певних відносинах з безлічі  $R$ . Виконаємо коротку інтерпретацію цих відносин. Так,

безліч відносин є сукупністю

$$R = \langle R_1, R_2, R_3, R_4 \rangle, \quad (2.1)$$

де  $R_1$  - відношення складу (*частина і ціле, приватне і загальне*),  $R_2$  -

відношення визначення ( $\epsilon$ ),  $R_3$  - відношення типу атрибута (*який*),  $R_4$

- відношення типу дії (*складає, призначений для, виконує, формує, ...*). Зазначений перелік відносин не є повним, проте він може бути розширений безпосередньо в ході розробки конкретної онтології. Зазначені відносини дозволяють безпосередньо формувати питання і генерувати відповіді. Якщо використовується такий тестування, то неправильні відповіді можуть генеруватися з відповідних складових інших концептів.

Для того, щоб пояснити детальніше, розберемо даний підхід на прикладі блоків верхнього рівня «Поняття» та «Означення». В останньому блоці необхідно виділити такі логічні атрибути, як: «базова компонента» ( $BC$ ), «уточнююча компонента» ( $SC$  – specified component), блоку «Поняття» присвоюємо атрибут – «назва означуваного поняття» ( $NmC$ ), а відношенню між

цими блоками – «зв’язок» ( $CNJ$  - conjunctive), «тип зв’язку» ( $TCNJ$ ). Завдяки деревовидній структурі вирішується проблема, що при безпосередньому використанні даної структури даних, завдяки відмінній кількості уточнюючих компонент, у різних означеннях виникає надлишковість подання. Доцільно об’єднати логічні атрибути «назва означуваного поняття», «базова компонента», та «уточнююча компонента» одним атрибутом «компонента твердження» ( $CS$ ). Крім того, мають бути використані поля для зображення деревовидної структури, тобто ідентифікатор елемента дерева ( $IE$ ) і посилання на батьківський елемент дерева ( $Prn_{IE}$ ). Таким чином отримуємо відношення, яке відображає текстове означення  $Df_i$  визначеної структури з довільною кількістю компонент:

$$Df_i = \{ [IE_{i,j}, CS_{i,j}, CNJ_{i,j}, TCNJ_{i,j}, Prn_{IE_{i,j}}] \}, \quad (2.2)$$

$$i = \overline{1, ND}, j = \overline{1, NC_i}, \quad (2.3)$$

де  $ND$  – кількість означень конкретного об’єкта,  $NC_i$  – кількість компонент  $i$ -го означення.

При цьому кожне означення зображаємо деревовидною структурою, у якій виділяємо записи для подання назв означуваних понять (3.4), базових (3.5)

$$(Prn_{IE_i} = 0) \Rightarrow (CS_i \in \{NmC\}) \cap (IE_i \in \{IE_0\}) \cap (TCNJ_i \neq NULL), \quad (2.4)$$

$$(Prn_{IE_{i,j}} \in \{IE_0\}) \Rightarrow (CS_{i,j} \in \{BC\}), \quad (2.5)$$

де  $\{NmC\}$  – множина назв означуваних понять семантичного об’єкта,  $\{IE_0\}$  – множина ідентифікаторів назв означуваних понять,  $\{BC\}$  – множина назв базових понять.

Отже, нульовий ідентифікатор батьківського елемента в співвідношенні (3.4) показує, що компонента цього елемента є назвою означуваного поняття, його ідентифікатор потрапляє у множину  $\{IE_0\}$ , і тип зв’язку в реченні має бути

визначеним. Якщо значення ідентифікатора батьківського елемента належить множині  $\{IE_0\}$ , то компонента цього елемента є базовим поняттям твердження, тобто належить множині  $\{BC\}$ .

На основі бази тверджень можемо сформуванати набір тестових завдань із заданими параметрами у вигляді динамічної структури. Під обсягом тесту розуміємо кількість тестових завдань  $NTT$ , а під параметрами тестового завдання – максимальну кількість альтернатив  $m_a$ , кількість правильних альтернатив  $t_a$  та рекомендовану кількість компонент основи тестового завдання  $k$ . Динамічна природа тестового завдання підвищує його варіативність та забезпечує автоматичну генерацію його ключа.

Структура тестового завдання  $TT_i^{k,m_a,t_a}$  містить дві компоненти – основу  $BTT_i^k$  та множину альтернатив  $SATT_i^{m_a,t_a}$ :

$$TT_i^{k,m_a,t_a} = [BTT_i^k, SATT_i^{m_a,t_a}], \quad (2.6)$$

$$BTT_i^k = \{[IBC_{i,j}, IE_{i,j}, CSI_{i,j}, CNJ_{i,j}, TCNJ_{i,j}, Prn_{IE_{i,j}}]\}, \quad (2.7)$$

$$i = \overline{1, NTT}, j \leq k, \quad (2.8)$$

де  $IBC_{i,j}$  – ідентифікатор компоненти основи тестового завдання.

Множина альтернатив тестового завдання складається з елементів  $ATT_{i,l}^{m_a,t_a}$ :

$$SATT_i^{m_a,t_a} = \{ATT_{i,l}^{m_a,t_a}\} \quad (2.9)$$

$$ATT_{i,l}^{m_a,t_a} = \{[IAC_{l,j}, IEA_{l,j}, IE_{l,j}, CS_{l,j}, CNJ_{l,j}, TCNJ_{l,j}, Prn_{IE_{l,j}}]\}, \quad (2.10)$$

$$l \leq m_a, k \leq j \leq NC_l \quad (2.11)$$

де  $IAC$  – ідентифікатор компоненти альтернативи тестового завдання,  $IEA_{l,j}$

–

ідентифікатор компоненти альтернативи у структурі означень (2.2),  $l$  позначає

номер означення, частиною якого є альтернатива. Якщо  $l = i$ , то альтернатива правильна, якщо  $l \neq i$  – альтернатива неправильна. При цьому для включення у тестове завдання батьківська компонента альтернативи у відношенні (2.2) повинна мати той же тип сполучника, що і остання компонента відповідного тестового завдання у відношенні (2.7).

Після проведення тестового контролю вибрані альтернативи можемо перевірити на достовірність за допомогою такого правила. Якщо вміст вказівника на батька першої компоненти альтернативи співпадає з ідентифікатором елемента означення останньої компоненти відповідної основи тесту, то альтернатива правильна, в іншому випадку вважаємо її неправильною. При цьому перша компонента альтернативи ідентифікується за тим, що значення її вказівника не дорівнює ідентифікатору попереднього елемента, а остання компонента основи тесту ідентифікується за тим, що вона є останньою у структурі, тобто значення вказівника наступної компоненти не співпадає з її ідентифікатором.

Це був деталізований принцип генерації питань одиничного або множинного вибору.

Для генерації тестових завдань альтернативного типу використовується алгоритм, що є модифікацією попередньо представленого:

1. Обрати два зв'язні блоки онтології;
2. Зафіксувати перший блок як основу запитання тесту та тип зв'язку між блоками;
3. випадковим чином обрати другу частину основи запитання тесту. При її істинності вона вибирається із другого блоку з обраних, а за хибності – з інших подібних блоків (обов'язковою умовою таких блоків є те, що тип зв'язку між подібними блоками має абсолютно співпадати з типом зв'язку обраних блоків).

Для генерації тестових завдань на встановлення відповідності використовується модифікований алгоритм для завдань множинного типу із деякими особливостями, а саме:

1. Обираємо множину подібних пар зв'язних блоків онтології;
2. Обираємо до кожного першого блоку єдине значення другого

(наприклад, якщо поняття має декілька означень, то потрібно вибрати одне з них) і записуємо їх по стовпчикам: в один – перші блоки, в інший - другі ;

3. Випадковим чином переставляємо елементи кожного із стовпчиків.

Метод	Критерії оцінювання					
	Технологія формування	Витрати	Пере-вірка теор. знань	Якість тестових завдань і їх лексична зрозумілість	Мінім. педагогічна цінність	Контроль складності
«Ручне» створення тестів	Ручне наповнення	Дуже високі, ручне формулювання	+	Висока, так як їх формування здійснюється безпосередньо людиною.	достатня	+
Параметризовані тести	Ручне наповнення, поповнене параметрами	Високі, ручне формулювання основ тестів	-	Середня, оскільки може бути не достатньо інформаційна умова завдання	достатня	+
Семантичні мережі	Необхідна повна формалізація предметної області	дуже високі (повна формалізація предметної області)	+	Завдання часто важкі для сприйняття людиною (признач. Для спілкування з машиною)	недостатня	-
Понятійно-тезисна модель	нерегламентоване ділення семантичних одиниць	невисокі (проте можливі, низькоякісні і альтернативні) порівняно	+	Достатньо висока (гірше ніж вручну, але ґрунтується на тексті викладача) Достатньо	недостатня	-

Система семантичних класів	регламентоване виділення семантичних одиниць	невисокі, (побудова семантичних класів)	+	висока (гірше ніж вручну, але ґрунтується на тексті викладача)	достатня	+
Онтологія	виділення зв'язних і подібних блоків онтології	Порівняно невисокі (побудова онтології навчального курсу)	+	Майже висока (гірше ніж вручну, але ґрунтується на тексті викладача і охоплює повний курс)	достатня	+

Таблиця 2.1 – Порівняння методів генерації тестових завдань

За такими принципами формуються питання використовуючи всі блоки предметної області. Доцільніше проводити генерацію питань до кожної окремо взятої теми.

У таблиці 3.1 проведено порівняльний аналіз існуючих методів генерації тестових завдань.

## 2.4 Генерація тестування

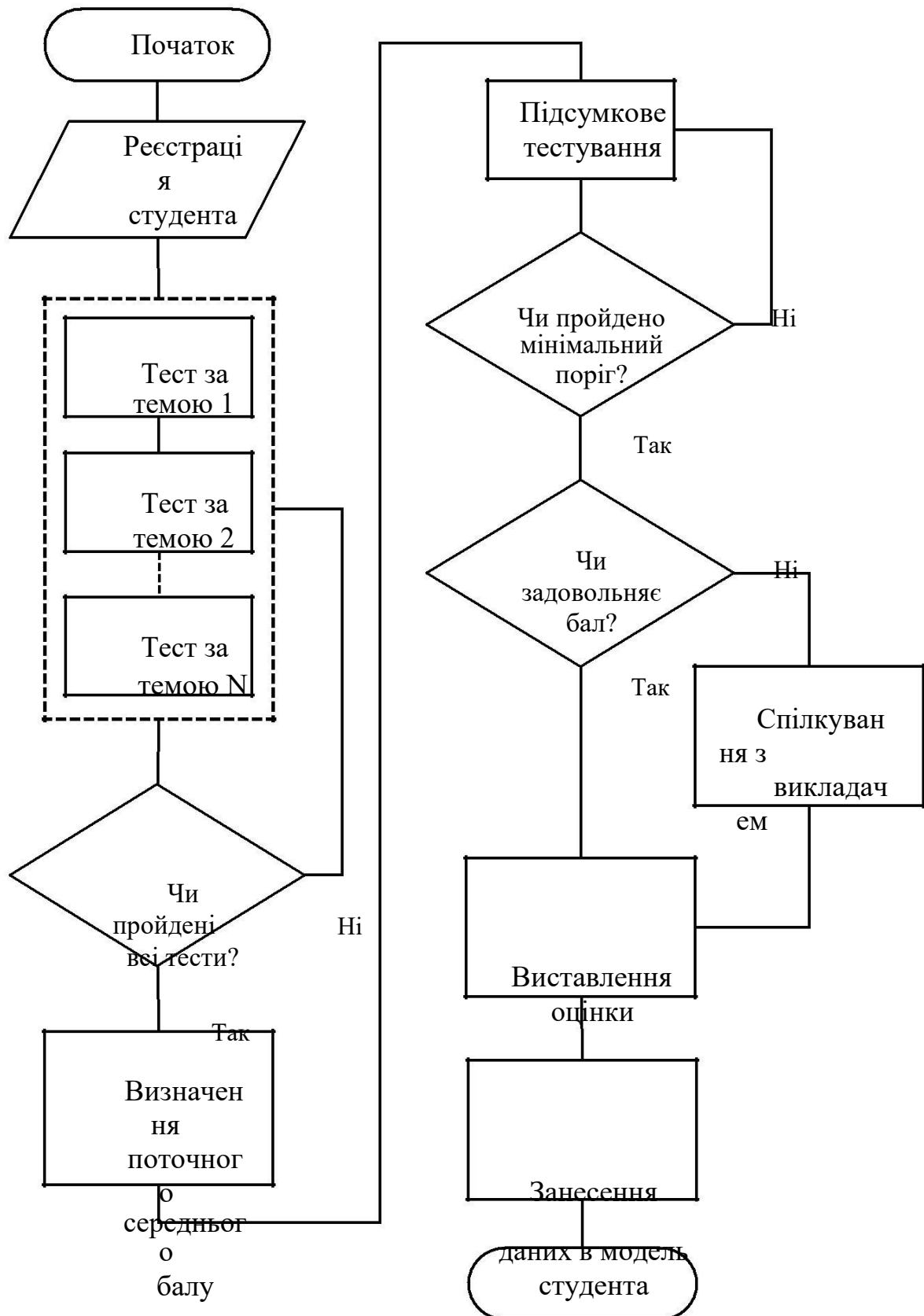
При проходженні навчальної дисципліни доцільніше проводити тестування не тільки по його закінченню, а й протягом усього навчання. Адже за один контроль знань неможливо в повній мірі виміряти рівень знань студента, а тим більш виставити оцінку. Тому пропонується такий алгоритм адаптивного контролю знань (рис 2.6).

Як видно із блок-схеми, спочатку потрібно провести окремі тестування до кожної теми. Це допоможе, по-перше, контролювати процес постійного навчання студентів, по-друге, визначити поточний середній бал, що відповідатиме рівню першого питання підсумкового тесту. Проходження усіх тематичних тестів є обов'язковим, адже поточний середній бал розраховується як середнє арифметичне за всі тести. Підсумкове тестування допоможе побачити загальну картину знань студента з даного предмету. Оскільки такий

тест дає підстави для виставлення підсумкової оцінки, проходження мінімального порогу є обов'язковим. Якщо студент пройшовши підсумкове тестування не погоджується з оцінкою, йому надається спроба довести свої знання і підвищити оцінку за допомогою спілкування з викладачем.

Для тематичного тестування слід обрати пірамідальне тестування, адже тільки для нього не важливий початковий рівень знань – воно завжди починається з середнього рівня важкості.





Кінець

Рисунок 2.6 – Алгоритм адаптивного контролю знань

А для підсумкового тестування пропонується модифікований метод комп'ютерного адаптивного тестування (КАТ). Як і в *stradaptive*-тестуванні нам

необхідні питання поділені на рівні складності. Оскільки теми можуть містити у собі різну кількість блоків в онтології предметної області, прийнято рішення зробити меншу, але однакову кількість рівнів складності - 6. Така кількість пов'язана з тим, що, хоча студент протягом вивчення і може набрати 100 балів,

виставляється буква. А їх кількість саме шість: А, В, С, D, E, F.

Наступне, що потрібно визначити, - звідки починається тестування. Як зазначалося раніше, студент проходить тематичні тести, після чого рахується середній поточний бал. Тож за цим балом і приймається рішення, з якого рівня починатиме тестування студент у підсумковому тестуванні. Це нагадує *flexilevel*-тестування.

Далі розподіляємо наші рівні складності на дві складові: верхні – А, В, С; нижні – D, E, F. Такі складові утворилися проведенням серединної межі усіма між рівнями. Ця межа відповідає за перехід від правильних відповідей до неправильних і навпаки. Що це означає? Коли студент знаходиться на верхніх рівнях і відповідає на питання правильно, то наступне питання дається з вищого рівня:  $C \rightarrow B, B \rightarrow A, A \rightarrow$  кінець тестування (з відміткою «відмінно»).

Знаходячись на нижніх рівнях і відповідаючи неправильно, наступне питання дається з нижчого рівня:  $D \rightarrow E, E \rightarrow F, F \rightarrow$  кінець тестування (з відміткою «не задовільно»). Складніша ситуація з протилежними відповідями. Якщо студент знаходиться на верхніх рівнях В і С, то відповівши на питання цих рівнів неправильно, наступне питання буде нижнього рівня D, а відповівши неправильно на питання рівня А – верхнього рівня С. Дзеркальна ситуація з нижнім рівнем. Коли студент правильно відповідає на питання нижнього рівня D і E, то наступне питання буде верхнього рівня С, а при правильній відповіді

на питання рівня F – нижнього рівня C. Схематично це виглядатиме, як зображено на рис. 2.7

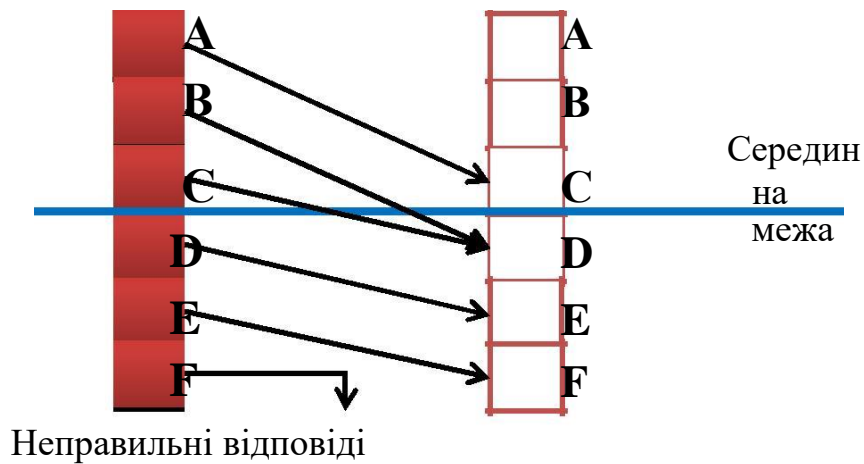
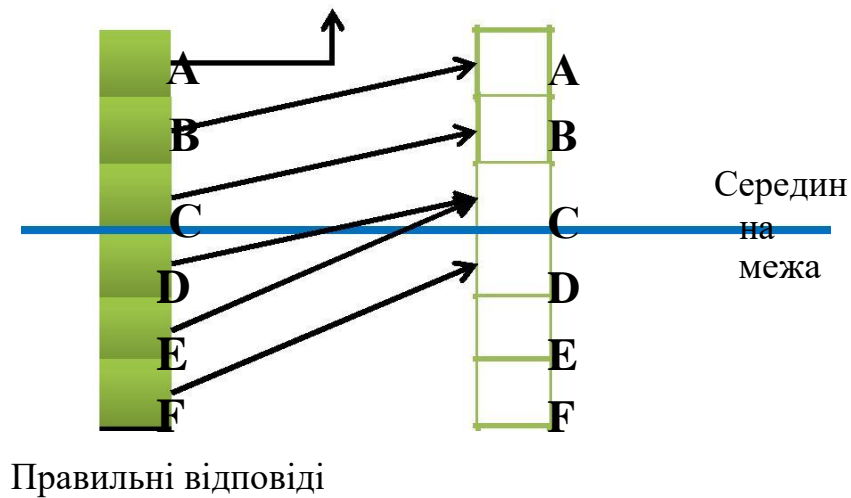


Рисунок 2.7 – Схема модифікованого КАТ

Така схема дозволить трохи вирівняти шанси між студентами. Адже досить багато ситуацій може виникнути при вивченні предмету. Наприклад такі: не вчив весь час, а перед підсумковим надолужив згаяне; чи навпаки, – весь семестр вчився, а під кінець розслабився; на більш складні питання знає відповідь, а базові (наприклад, визначення) не знає і т.д. А також значно пришвидшить час проходження тесту для студентів, які відмінно знають предмет і які взагалі не знають.

## 2.5 Алгоритм вибору навчального матеріалу

Процес оцінювання рівня знань студента потрібно, щоб як найкраще вибрати алгоритм для його подальшого навчання. Контроль знань, в свою чергу, проводиться з різною метою: для попереднього оцінювання, для визначення проміжних результатів, для результуючого оцінювання і т. д. За його результатами приймається рішення про доцільність подальшого навчання, підвищення рівня знань, припинення навчання.

Елементарною одиницею є блок навчального матеріалу (БНМ). У разі навчання кожного конкретного фахівця необхідно визначити план його навчання за результатами контролю знань. [22]

Структурну схему контролю знань яку запропонували Снитюк В.Е., Юрченко К.Н. у своїй монографії «Інтелектуальне управління оцінюванням знань», зображена на рис. 2.8

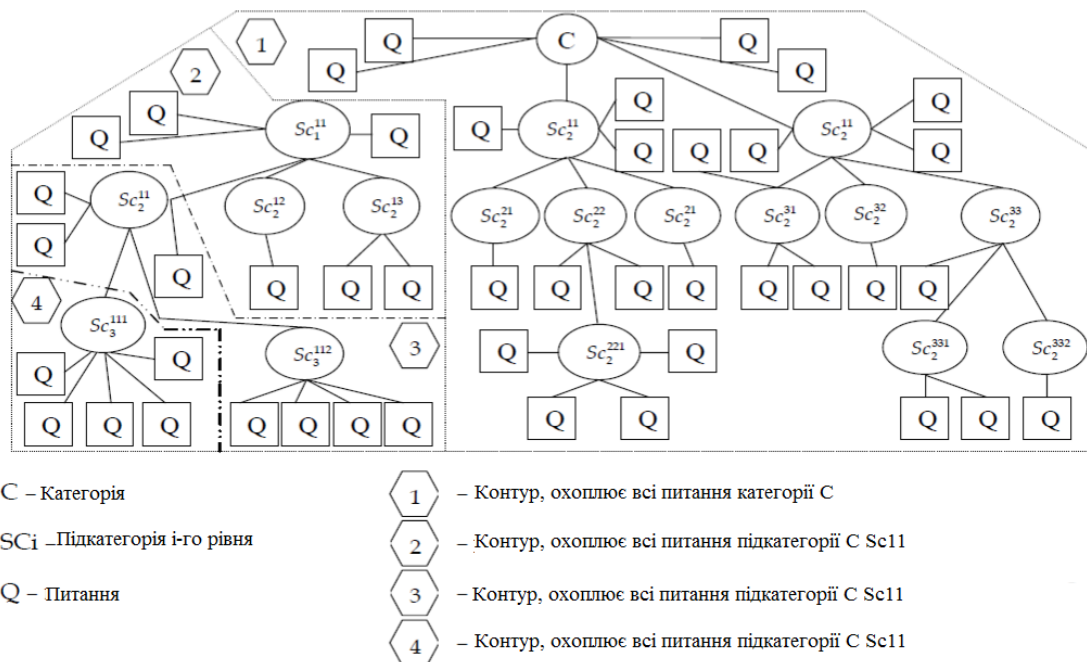


Рис2.8 Логічна схема контролю знань

З малюнку можна побачити, що сама структура має ієрархічний вигляд, в якій є категорії і підкатегорії. Самі питання відносяться або до самої категорії,

або ж до категорії відповідно.

Алгоритм побудови навчання студентів побудовано наступним чином. Розглянемо такі приклади. Студент проходить вхідний тест (вхідний контроль) по певній дисципліні, або групі дисциплін. В залежності від того як він пройде цей вхідний тест буде будуватися його навчання. У своїй роботі я пропоную використовувати наступну схему.

Розіб'ємо наш вхідний тест по будь-якій дисципліні на п'ятибальну шкалу, як це показано в таблиці 2.2.

Оцінка	Знання
4-5	Учень знає предмет
3	Деякі знання присутні
1-2	Зовсім не знає предмет

Таблиця 2.2 – Умови подачі навчального матеріалу.

Як ми бачимо з таблиці я пропоную будувати процес навчання так. Якщо студент або учень, проходить вхідний тест і отримує за нього 4-5 балів, то можна стверджувати, що студент знає цей предмет, і процес його навчання має складатися більш загально, тобто подача блоку навчального матеріалу має бути більшою і загальнішою, оскільки ціль цього випадку є не навчити, а скоріше освіжити пам'ять і підтримати рівень знань.

Якщо ж студень отримав після проходження вхідного контролю 3, то можна стверджувати що базові знання, або розуміння предмету є, але вони є не достатніми. Тому в цьому випадку я пропоную надавати блоки навчального матеріалу, більш детальніше, роз'яснюючи певні моменти, наводити приклади, щоб поступово студент міг опанувати предмет.

Що стосується тих хто отримав 1-2 бали, тут можна говорити, що або цей

предмет не вивчався студентом взагалі, або вивчався, але досить погано. В цьому випадку я пропоную розбивати блоки навчального матеріалу дуже детально, і роз'яснювати всі моменти дисципліни, оскільки я вважаю що не можна опанувати предмет, якщо не розуміти тривіальні речі. Тобто ми не можемо навчити студента писати віршів, якщо він не знає букв. Так і в цьому випадку, потрібно пояснити прості речі, щоб далі можна було зрозуміти увесь матеріал, і отримати задовільну оцінку.



## 2.6 Висновки

Сучасні дослідження в галузі тестування і оцінювання показали, що потенціал комп'ютерних адаптивних тестів постійно збільшується. Переваги комп'ютерного адаптивного тестування дають можливість зробити крок до розвитку тестування в майбутньому. Проте робити такий крок завжди треба виважено для того, щоб така процедура оцінювання добре інтегрувалася в процес навчання для забезпечення його максимальної ефективності.

Обґрунтовано актуальність досліджень в сфері автоматичної генерації тестових завдань, яка спрямована на економію людських ресурсів, проаналізовано основні відомі підходи, визначено основні напрями розвитку даної проблематики.

Запропонована система генерації питань, та алгоритм подач навчального матеріалу. Показано можливості використання її для складання тестових питань закритого типу, які дозволяють отримувати завдання достатньо високої складності та носять педагогічну цінність. Це полегшить сам принцип, а в подальшому реалізацію такої системи. Крім цього ця система предметного курсу може застосовуватись і в інших аспектах, що робить її універсальною.

Побудовано алгоритм адаптивного контролю знань, який включає як тематичні тестування (тобто окремий тест для кожної теми), так і підсумкове. Ця система зосереджена тільки на тестовому контролі знань. Звичайно, не можна виставити оцінку, опираючись на дані тестування, але слід обов'язково її враховувати.

Розроблено новий варіант тестування, в основу якого покладено 6 рівнів складності і серединна межа. Хоча він і трохи складніший від вже існуючих, але зберігає всі параметри адаптивного тестування і є ефективним та гарно підлаштовується під студентів, порівнюючи їх шанси на підсумковому тестуванні.

### 3. Програмна реалізація, її тестування та результати

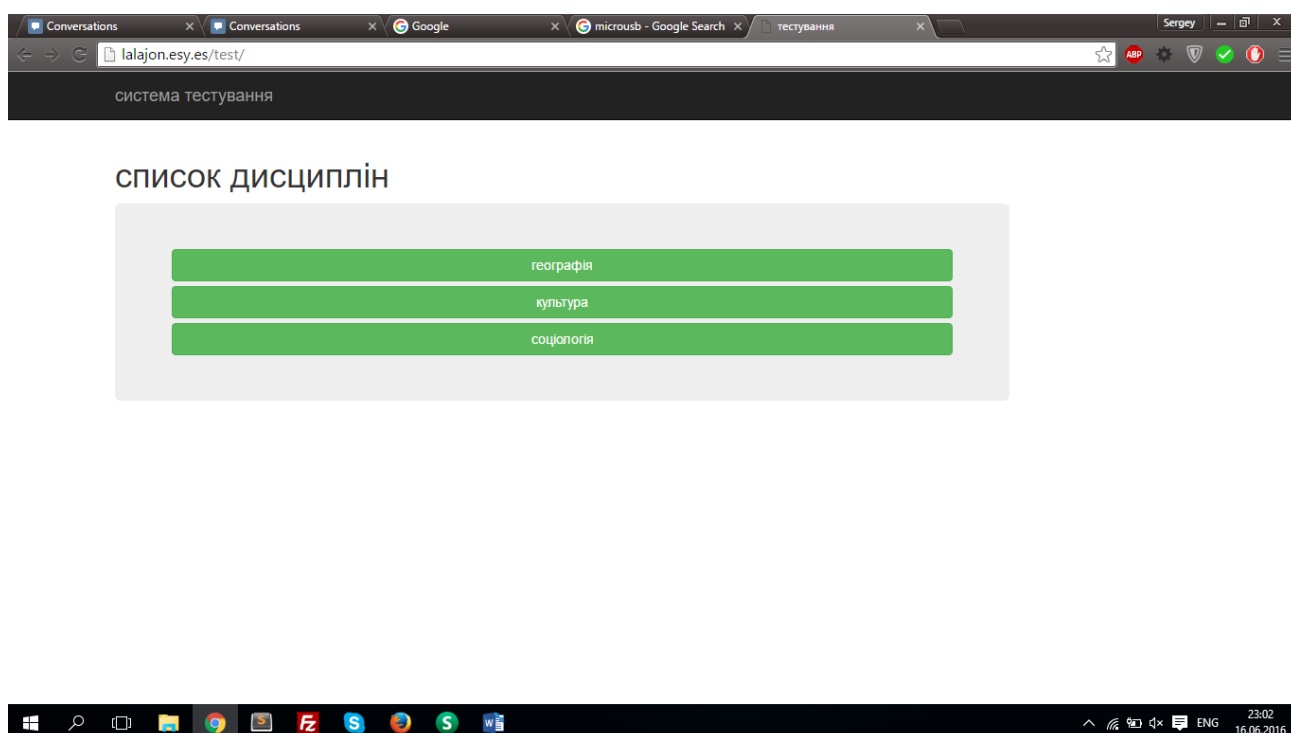
У даному розділі будуть наведені результати роботи розробленого програмного продукту, будуть зроблені відповідні висновки.

#### 3.1 Аналіз роботи програмного продукту

Технології дистанційного і електронного навчання здатні значно вплинути на підвищення якості і ефективності процесу підготовки висококваліфікованих працівників.

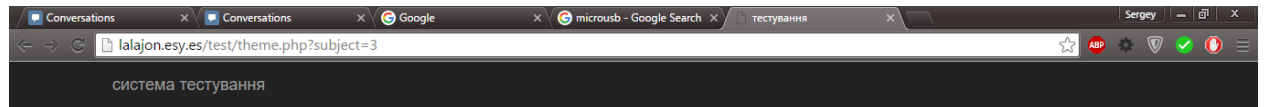
Систему навчання і контролю знань яка була мною запропонована і реалізована, задовольняє більшість вимог, що ставляться користувачами до систем електронного навчання, використовують також для проведення комп'ютерного тестування. Розглянемо саму реалізацію.

Після того як студент пройшов авторизацію йому буде запропоновано список дисциплін (Матеріали і тести до курсу, можна постійно оновлювати).

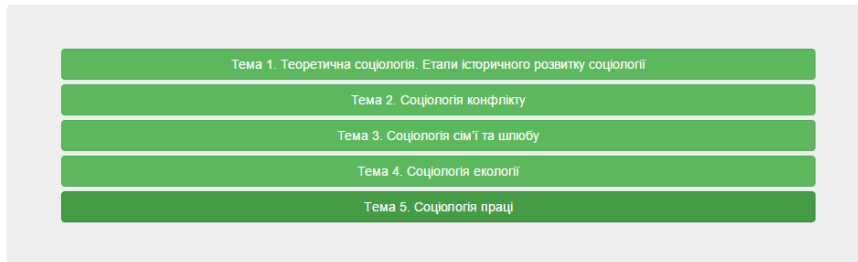


### Рисунок 3.1 – Список дисциплін

При виборі дисципліни ми переходимо на наступну сторінку із темами які доступні студентам.

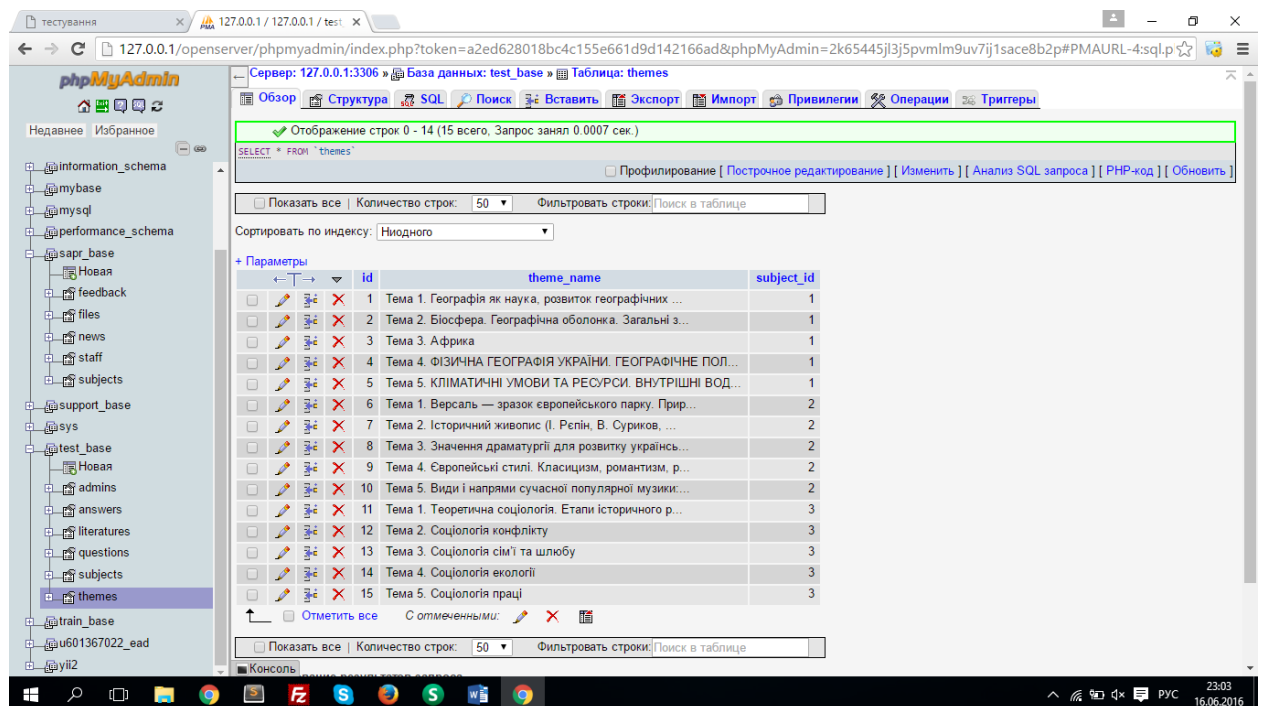


тести доступні за наступними темами



### Рисунок 3.2 – Тести доступні за наступними темами

Всі теми зберігаються у базі даних.



### Рисунок 3.3 – База даних(навчальні блоки)

У кожній темі є свій id. Тобто при виборі курсу із бази даних вибирається вибірка саме потрібних нам матеріалів.

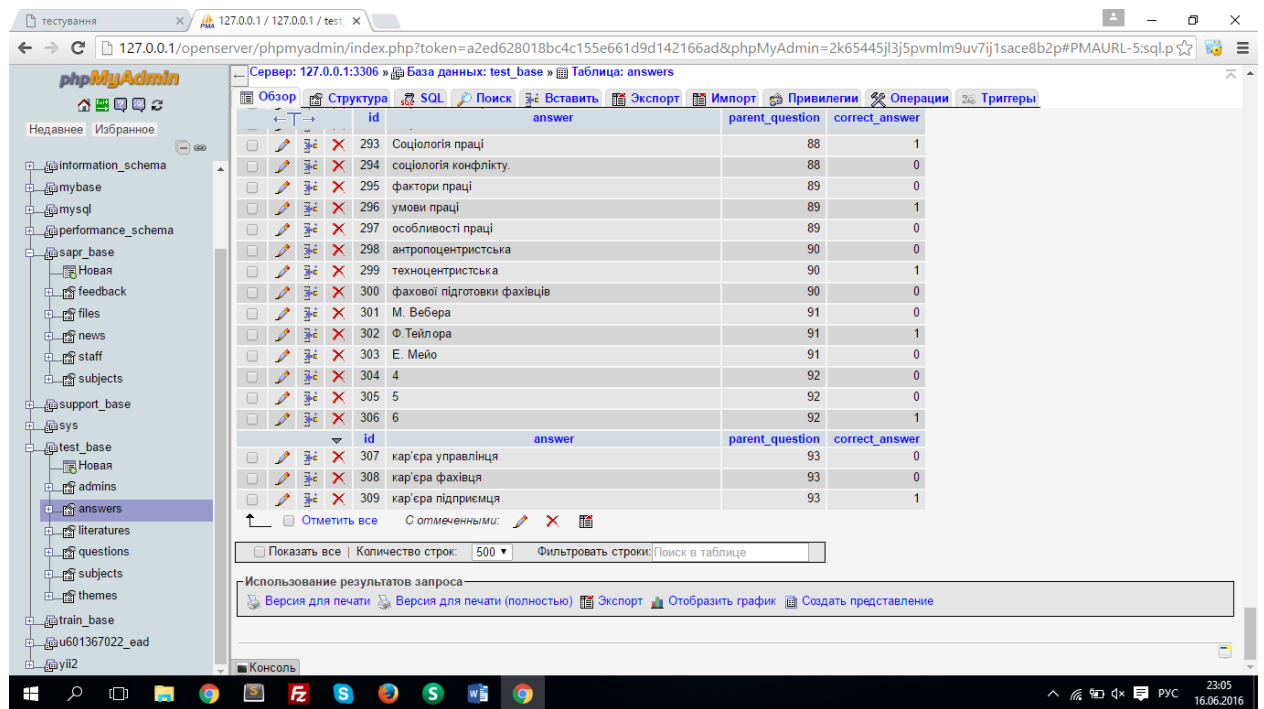


Рисунок 3.4 – База даних(тестові запитання)

Те саме відбувається і з відповідями на запитання. Також ми можемо побачити складну структуру проектування – ієрархії. Кожна табличка зв'язана з іншою через id.

Після того, як студент вибрав дисципліну і тему йому пропонується пройти вхідний тест.



питання номер 5: "Найбільш високорослі народи Африки — пілоти, тутсі, хіна — проживають у саванах північної частини материка. Їхній середній зріст становить"

- 160-170 см
- 150-160 см
- 180 – 200 см



Рисунок 3.5 – Вхідний тест

Питання йдуть один-за-одним і паралельно йде підрахунок правильних відповідей.

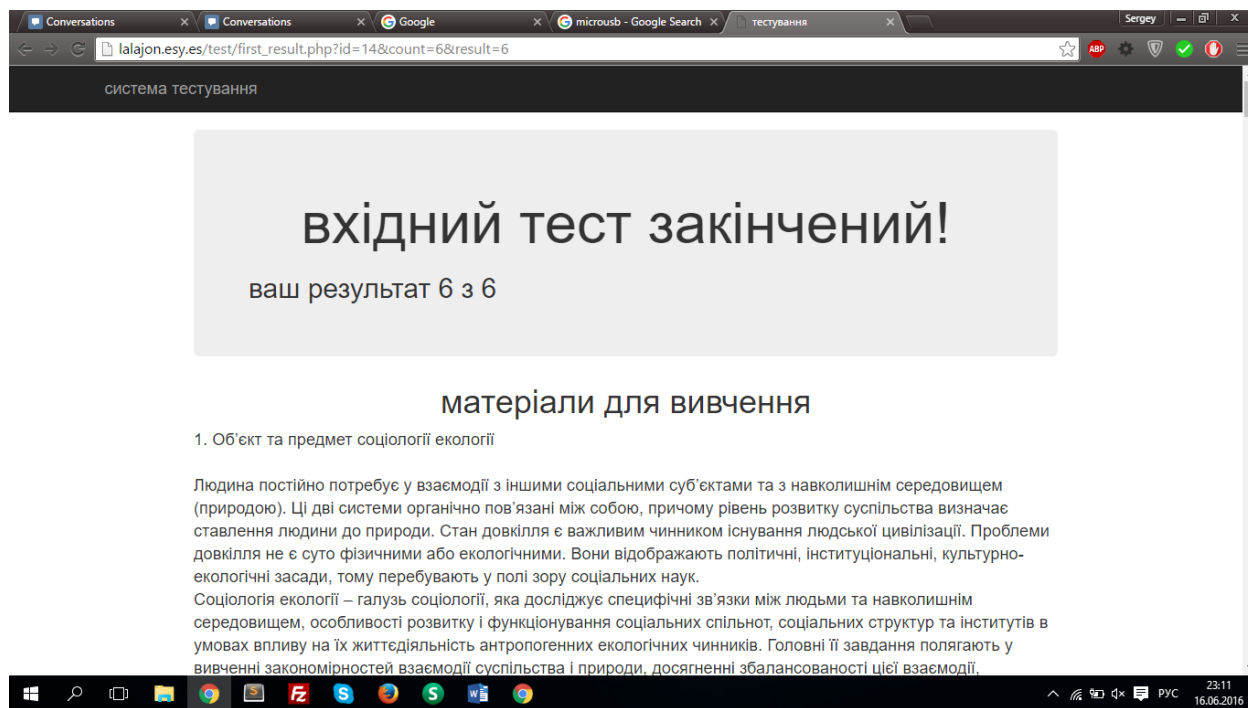


Рисунок 3.6 – Вхідний тест

Після того, як був складений вхідний тест, система оцінює рівень знань студента і пропонує матеріали для вивчення. В залежності від того яку оцінку студент отримав подача навчального матеріалу відрізняється.

Також система рекомендує літературу в залежності від проходження вхідного тесту.

## 3.2 Висновки

Для процесу контролю знань характерна значна суб'єктивність, об'єктивізувати яку прагнуть багато викладачів і екзаменованих. Необхідно зазначити, що такі процеси є різноаспектними, напрямки дослідження базуються на різних концептуальних парадигмах. У незначній мірі відбувається об'єктивізація процесу оцінювання. Разом з тим, втрачається повнота охоплення

навчального матеріалу і різноманітність, яке виражається в семантичній сутності задаються. Контроль знань у більшості випадків не оптимізований за часом проведення і смисловій завантаженості питань.

Запропонована система генерації питань, що ґрунтується на навчального матеріалу. Показано можливості використання її для складання тестових питань закритого типу, які дозволяють отримувати завдання достатньо високої складності та носять педагогічну цінність. Це полегшить сам принцип, а в подальшому реалізацію такої системи.

Побудовано алгоритм адаптивного контролю знань, який включає як тематичні тестування (тобто окремий тест для кожної теми), так і підсумкове. Ця система зосереджена тільки на тестовому контролі знань. Звичайно, не можна виставити оцінку, опираючись на дані тестування, але слід обов'язково її враховувати.

## 4. ФУНКЦІОНАЛЬНО-ВАРТІСНИЙ АНАЛІЗ ПРОГРАМНОГО ПРОДУКТУ

### 4.1 ВСТУП

У даному розділі проводиться оцінка основних характеристик програмного продукту, призначеного для контролю знань студента і його навчання, за допомогою адаптивних технологій.

Програмний продукт був розроблений за допомогою мови програмування PHP – є однією з найпоширеніших мов, що використовуються у сфері веб-розробки та MySQL — вільна система керування реляційними базами даних.

Програмний продукт є крос-платформенним та рекомендується для використання на персональних комп'ютерах під управлінням операційних систем Windows, Linux чи Mac.

Нижче наведено аналіз різних варіантів реалізації модулю з метою вибору оптимальної, з огляду при цьому як на економічні фактори, так і на характеристики продукту, що впливають на продуктивність роботи і на його сумісність з апаратним забезпеченням. Для цього було використано апарат функціонально-вартісного аналізу.

Функціонально-вартісний аналіз (ФВА) – це технологія, яка дозволяє оцінити реальну вартість продукту або послуги незалежно від організаційної структури компанії. Як прямі, так і побічні витрати розподіляються по продуктам та послугам у залежності від потрібних на кожному етапі виробництва обсягів ресурсів. Виконані на цих етапах дії у контексті метода ФВА називаються функціями.

Мета ФВА полягає у забезпеченні правильного розподілу ресурсів, виділених на виробництво продукції або надання послуг, на прямі та непрямі витрати. У даному випадку – аналізу функцій програмного продукту й виявлення усіх витрат на реалізацію цих функцій.

Фактично цей метод працює за таким алгоритмом:

– визначається послідовність функцій, необхідних для виробництва продукту. Спочатку – всі можливі, потім вони розподіляються по двом групам: ті, що впливають на вартість продукту і ті, що не впливають. На цьому ж етапі оптимізується сама послідовність скороченням кроків, що не впливають на цінність і відповідно витрат;

– для кожної функції визначаються повні річні витрати й кількість робочих часів;

– для кожної функції на основі оцінок попереднього пункту визначається кількісна характеристика джерел витрат;

– після того, як для кожної функції будуть визначені їх джерела витрат, проводиться кінцевий розрахунок витрат на виробництво продукту.

## **4.2 ПОСТАНОВКА ЗАДАЧІ**

У роботі застосовується метод ФВА для проведення техніко-економічного аналізу розробки призначеного для контролю знань студента і його навчання, за допомогою адаптивних технологій. Оскільки основні проектні рішення стосуються всієї системи, кожна окрема підсистема має їм задовольняти. Тому фактичний аналіз представляє собою аналіз функцій програмного продукту, призначеного для створення бази знань наукових публікацій з використанням RDF-сховищ.

Відповідно цьому варто обирати і систему показників якості програмного продукту.

Технічні вимоги до продукту наступні:

– програмний продукт повинен функціонувати на персональних комп'ютерах із стандартним набором компонент;

– забезпечувати високу швидкість обробки великих об'ємів даних у реальному часі;



– забезпечувати зручність і простоту взаємодії з користувачем або з розробником програмного забезпечення у випадку використання його як модуля;

– передбачати мінімальні витрати на впровадження програмного продукту.

#### **4.2.1 ОБҐРУНТУВАННЯ ФУНКЦІЙ ПРОГРАМНОГО ПРОДУКТУ**

Головна функція  $F_0$  – розробка бази знань наукових публікацій. Виходячи з конкретної мети, можна виділити наступні основні функції ПП:

$F_1$  – вибір середовища проектування;

$F_2$  – використання існуючих онтологій;

$F_3$  – спосіб пошуку у базі знань.

Кожна з основних функцій може мати декілька варіантів реалізації.

Функція  $F_1$ :

а) середовище проектування NetBeans;

б) середовище проектування MySQL;

Функція  $F_2$ :

а) створення своєї онтології;

б) доповнення існуючої онтології;

Функція  $F_3$ :

а) пошук за бібліометричними даними;

б) пошук за контекстом, ключовими словами.

#### **4.2.2 ВАРІАНТИ РЕАЛІЗАЦІЇ ОСНОВНИХ ФУНКЦІЙ**

Варіанти реалізації основних функцій наведені у морфологічній карті системи (рис. 4.1). На основі цієї карти побудовано позитивно-негативну матрицю варіантів основних функцій (таблиця 4.1).

Морфологічна карта відображує всі можливі комбінації варіантів реалізації функцій, які складають повну множину варіантів ПП.

На основі аналізу позитивно-негативної матриці робимо висновок, що при розробці програмного продукту деякі варіанти реалізації функцій варто

відкинути, тому, що вони не відповідають поставленим перед програмним продуктом задачам.



Рисунок 4.1 – Морфологічна карта

Таблиця 4.1 – Позитивно-негативна матриця

Основні функції	Варіанти реалізації	Переваги	Недоліки
<i>F1</i>	<i>A</i>	Займає менше часу при проектуванні онтології і створенні бази знань	Використання більше ресурсів системи
	<i>B</i>	Використання менше ресурсів системи	Займає більше часу при проектуванні онтології і створенні бази знань
<i>F2</i>	<i>A</i>	Онтологія відповідає всім поставленим задачам	Займає більше часу при проектуванні
	<i>B</i>	Займає менше часу при проектуванні	Необхідність скоректувати для поставлених цілей

<i>F3</i>	<i>A</i>	Легша реалізація	Дає гірші результати
	<i>B</i>	Дає кращі результати	Важча реалізація

Функція *F1*:

Оскільки потрібно спроектувати онтологію і заповнити великою кількістю публікацій, тому потрібно швидко це зробити, тому варіант б) має бути відкинтий.

Функція *F2*:

Оскільки корекція існуючої бази знань та створення власної, вважаємо варіанти а) та б) гідними розгляду.

Функція *F3*:

Кращі результати(релевантний пошук) є важливішим для даної роботи, ніж складність реалізації, тому варіант а) має бути відкинтий.

Таким чином, будемо розглядати такі варіанти реалізації ПП:

1. *F1a* – *F2a* – *F3б*
2. *F1a* – *F2б* – *F3б*

Для оцінювання якості розглянутих функцій обрана система параметрів, описана нижче.

## **4.3 ОБҐРУНТУВАННЯ СИСТЕМИ ПАРАМЕТРІВ ПП**

### **4.3.1 ОПИС ПАРАМЕТРІВ**

На підставі даних про основні функції, що повинен реалізувати програмний продукт, вимог до нього, визначаються основні параметри виробу, що будуть використані для розрахунку коефіцієнта технічного рівня.

Для того, щоб охарактеризувати програмний продукт, будемо використовувати наступні параметри:

- *X1* – релевантність пошукових результатів;

- X2 – об’єм пам’яті для збереження даних;
- X3 – час обробки даних;
- X4 – потенційний об’єм програмного коду.

X1: Відображає відповідність результатів пошуку операцій залежно від обраної архітектури онтології.

X2: Відображає об’єм пам’яті в оперативній пам’яті персонального комп’ютера, необхідний для збереження та обробки даних під час виконання програми.

X3: Відображає час, який витрачається на дії.

X4: Показує розмір програмного коду який необхідно створити безпосередньо розробнику.

#### 4.3.2 КІЛЬКІСНА ОЦІНКА ПАРАМЕТРІВ

Гірші, середні і кращі значення параметрів вибираються на основі вимог замовника й умов, що характеризують експлуатацію ПП як показано у табл. 9.

Таблиця 4.2 – Основні параметри ПП

Назва Параметра	Умовні позначення	Одиниці виміру	Значення параметра		
			Гірші	середні	кращі
Відповідність результатів пошуку	X1	Бали	1	3	5
Об’єм пам’яті для збереження даних	X2	Мб	32	16	8
Час обробки запиту до бази знань	X3	мс	600	320	75
Потенційний об’єм програмного коду	X4	кількість строк коду	1855	1325	985

За даними таблиці 3 будуються графічні характеристики параметрів – рис. 4.2 – рис. 4.4.

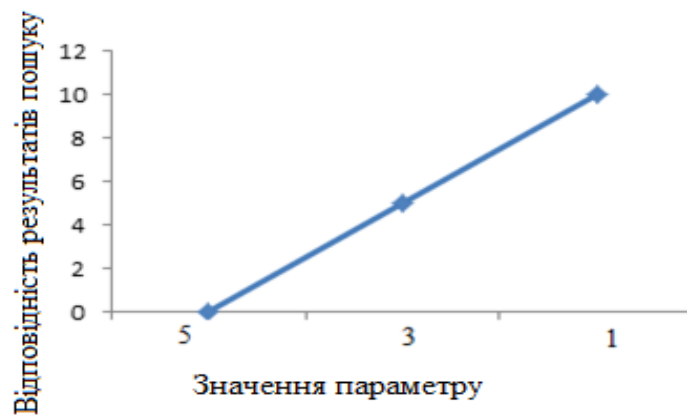


Рисунок 4.2 – Відповідність результатів пошуку

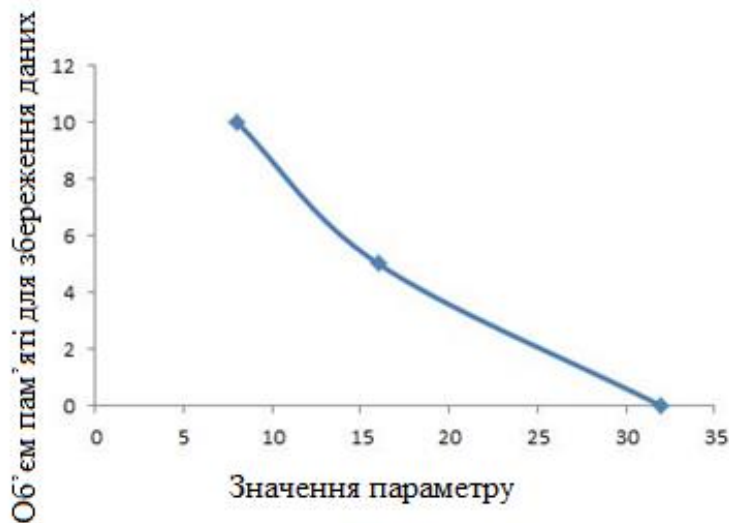


Рисунок 4.3 – Об'єм пам'яті для збереження даних

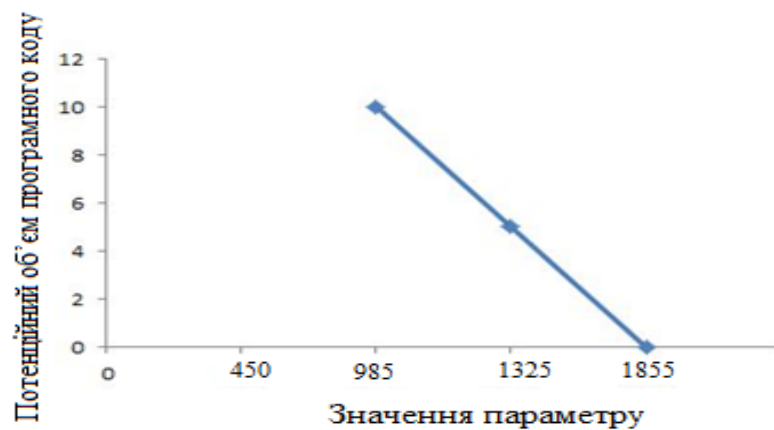


Рисунок 4.4 – Потенційний об'єм програмного коду

### 4.3.3 АНАЛІЗ ЕКСПЕРТНОГО ОЦІНЮВАННЯ ПАРАМЕТРІВ

Після детального обговорення й аналізу кожний експерт оцінює ступінь важливості кожного параметру для конкретно поставленої цілі – розробка

програмного продукту, який дає найбільш точні результати при знаходженні параметрів моделей адаптивного прогнозування і обчислення прогнозних значень.

Значимість кожного параметра визначається методом попарного порівняння. Оцінку проводить експертна комісія із 7 людей. Визначення коефіцієнтів значимості передбачає:

- визначення рівня значимості параметра шляхом присвоєння різних рангів;
- перевірку придатності експертних оцінок для подальшого використання;
- визначення оцінки попарного пріоритету параметрів;
- обробку результатів та визначення коефіцієнту значимості.

Результати експертного ранжування наведені у таблиці 3.

Для перевірки степені достовірності експертних оцінок, визначимо наступні параметри:

а) сума рангів кожного з параметрів і загальна сума рангів:

$$R_i = \sum_{j=1}^N r_{ij} R_{ij} = \frac{Nn(n+1)}{2} = 105,$$

де  $N$  – число експертів,  $n$  – кількість параметрів;

Таблиця 4.3 – Результати ранжування параметрів

Позначення параметра	Назва параметра	Одиниці виміру	Ранг параметра за оцінкою експерта							Сума рангів $R_i$	Відхилення $\Delta_i$	$\Delta_i^2$
X1	Відповідність результатів пошуку	Бали								27	0.75	0.56
X2	Об'єм пам'яті для збереження даних	Мб								25	-1.25	1.56
X3	Час обробки даних алгоритмом	Мс								12	-14.25	203.06



X3 і X4	>	>	>	>	>	>	>	>	>	1.5
---------	---	---	---	---	---	---	---	---	---	-----

Числове значення, що визначає ступінь переваги  $i$ -го параметра над  $j$ -тим,  $a_{ij}$  визначається по формулі:

$$a_{ij} = \begin{cases} 1.5 \text{ при } X_i > X_j \\ 1.0 \text{ при } X_i = X_j \\ 0.5 \text{ при } X_i < X_j \end{cases}$$

З отриманих числових оцінок переваги складемо матрицю  $A = \| a_{ij} \|$ .

Для кожного параметра зробимо розрахунок вагомості  $K_{vi}$  за наступними формулами:

$$K_{vi} = \frac{b_i}{\sum_{i=1}^n b_i}, \text{ де } b_i = \sum_{i=1}^N a_{ij}.$$

Відносні оцінки розраховуються декілька разів доти, поки наступні значення не будуть незначно відрізнятися від попередніх (менше 2%). На другому і наступних кроках відносні оцінки розраховуються за наступними формулами:

$$K_{vi} = \frac{b'_i}{\sum_{i=1}^n b'_i}, \text{ де } b'_i = \sum_{i=1}^N a_{ij} b_j.$$

Як видно з таблиці 12, різниця значень коефіцієнтів вагомості не перевищує 2%, тому більшої кількості ітерацій не потрібно.

Таблиця 4.5 – Розрахунок вагомості параметрів

Параметри $x_i$	Параметри $x_j$				Перша ітер.		Друга ітер.		Третя ітер.	
	X1	X2	X3	X4	$b_i$	$K_{vi}$	$b_i^1$	$K_{vi}^1$	$b_i^2$	$K_{vi}^2$
X1	1	0	0	1	3.5	0.2	22.	0.2	100	0.2
	.0	.5	.5	.5		19	25	16		15
X2	1	1	0	1	4.5	0.2	27.	0.2	124	0.2
	.5	.0	.5	.5		81	25	82	.25	83
X3	1	1	1	1	5.5	0.3	34.	0.3	156	0.3



	.5	.5	.0	.5		44	25	47		48
X4	0	0	0	1	2.5	0.1	14.	0.1	64.	0.1
	.5	.5	.5	.0		56	25	55	75	54
Всього:					16	1	98	1	445	1

#### 4.4 АНАЛІЗ РІВНЯ ЯКОСТІ ВАРІАНТІВ РЕАЛІЗАЦІЇ ФУНКЦІЙ

Визначаємо рівень якості кожного варіанту виконання основних функцій окремо.

Абсолютні значення параметрів X2(об'єм пам'яті для збереження даних) X1 (відповідність результатів пошуку) та X3 відповідають технічним вимогам умов функціонування даного ПП.

Абсолютне значення параметра X4 (потенційний об'єм програмного коду) обрано не найгіршим (не максимальним), тобто це значення відповідає або варіанту а) 1800 або варіанту б) 1200.

Коефіцієнт технічного рівня для кожного варіанта реалізації ПП розраховується так (таблиця б):

$$K_K(j) = \sum_{i=1}^n K_{ei,j} B_{i,j},$$

де  $n$  – кількість параметрів;  $K_{ei}$  – коефіцієнт вагомості  $i$ -го параметра;  $B_i$  – оцінка  $i$ -го параметра в балах.

Таблиця 4.6 – Розрахунок показників рівня якості варіантів реалізації основних функцій ПП

Основні функції	Варіант реалізації функції	Абсолютне значення параметра	Бальна оцінка параметра	Коефіцієнт вагомості параметра	Коефіцієнт рівня якості
F1(X1)	А	11000	3.6	0.215	0.774
F2(X3)	А, Б	800	2.4	0.348	0.835
F2(X4)	А	1800	2	0.154	0.308
	Б	1200	8	0.154	1.232

F3(X2)	Б	16	3.4	0.283	0.962
--------	---	----	-----	-------	-------

За даними з таблиці 13 за формулою

$$K_K = K_{\text{ТУ}}[F_{1k}] + K_{\text{ТУ}}[F_{2k}] + \dots + K_{\text{ТУ}}[F_{zk}],$$

визначаємо рівень якості кожного з варіантів:

$$K_{K1} = 0.774 + 0.835 + 0.308 + 0.962 = 2.879$$

$$K_{K2} = 0.774 + 0.835 + 1.232 + 0.962 = 3.803$$

Як видно з розрахунків, кращим є другий варіант, для якого коефіцієнт технічного рівня має найбільше значення.

#### 4.5 ЕКОНОМІЧНИЙ АНАЛІЗ ВАРІАНТІВ РОЗРОБКИ ПП

Для визначення вартості розробки ПП спочатку проведемо розрахунок трудомісткості.

Всі варіанти включають в себе два окремих завдання:

1. Розробка проекту програмного продукту;
2. Розробка програмної оболонки;

При цьому варіант 3 має додаткове завдання:

3. Реалізація пошуку за контекстом;

А варіант 4 має інше додаткове завдання:

4. Обробка інтерфейсу готових бібліотек.

Завдання 1 за ступенем новизни відноситься до групи А, завдання 2 – до групи Б. За складністю алгоритми, які використовуються в завданні 1 належать до групи 1; а в завданні 2 – до групи 2.

Для реалізації завдання 1 використовується довідкова інформація, а завдання 2 використовує інформацію у вигляді даних.

Проведемо розрахунок норм часу на розробку та програмування для кожного з завдань. Загальна трудомісткість обчислюється як

$$T_0 = T_p \cdot K_{\text{П}} \cdot K_{\text{СК}} \cdot K_{\text{М}} \cdot K_{\text{СТ}} \cdot K_{\text{СТ.М}},$$

де  $T_p$  – трудомісткість розробки ПП;  $K_{\text{П}}$  – поправочний коефіцієнт;  $K_{\text{СК}}$  – коефіцієнт на складність вхідної інформації;  $K_{\text{М}}$  – коефіцієнт створення ієрархії

класів і відносин;  $K_{СТ}$  – коефіцієнт використання стандартних модулів і прикладних програм;  $K_{СТ.М}$  – коефіцієнт стандартного математичного забезпечення

Для першого завдання, виходячи із норм часу для завдань розрахункового характеру степеню новизни А та групи складності алгоритму 1, трудомісткість дорівнює:  $T_p = 90$  людино-днів. Поправочний коефіцієнт, який враховує вид нормативно-довідкової інформації для першого завдання:  $K_{П} = 1.7$ . Поправочний коефіцієнт, який враховує складність контролю вхідної та вихідної інформації для всіх завдань рівний 1:  $K_{СК} = 1$ . Оскільки при розробці першого завдання використовуються стандартні модулі, врахуємо це за допомогою коефіцієнта  $K_{СТ} = 0.8$ . Тоді загальна трудомісткість програмування першого завдання дорівнює:

$$T_1 = 90 \cdot 1.7 \cdot 0.8 = 122.4 \text{ людино-днів.}$$

Проведемо аналогічні розрахунки для подальших завдань.

Для другого завдання (використовується алгоритм другої групи складності, ступінь новизни Б), тобто  $T_p = 27$  людино-днів,  $K_{П} = 0.9$ ,  $K_{СК} = 1$ ,  $K_{СТ} = 0.8$ :

$$T_2 = 27 \cdot 0.9 \cdot 0.8 = 19.44 \text{ людино-днів.}$$

Для третього завдання (використовується алгоритм другої групи складності, ступінь новизни Г з використанням перемінної інформації):

$$T_p = 14 \text{ людино-днів;}$$

$$K_{П} = 0.72; K_{СТ} = 0.8;$$

$$T_o = 14 \cdot 0.72 \cdot 0.8 = 7.05.$$

Для четвертого завдання (використовується алгоритм третьої групи складності, ступінь новизни Г):

$$T_p = 9 \text{ людино-днів;}$$

$$K_{П} = 0.6; K_{СТ} = 1;$$

$$T_o = 9 \cdot 0.6 \cdot 1 = 5.4.$$

Складаємо трудомісткість відповідних завдань для кожного з обраних варіантів реалізації програми, щоб отримати їх трудомісткість:

$$T_I = (122.4 + 19.44 + 7.05) \cdot 8 = 1195 \text{ людино-годин};$$

$$T_{II} = (122.4 + 19.44 + 5.4) \cdot 8 = 1181.22 \text{ людино-годин};$$

Найбільш високу трудомісткість має варіант I.

В розробці беруть участь два програмісти із окладом 7000 грн., один фінансовий аналітик з окладом 10000 грн. Визначимо зарплату за годину за формулою:

$$C_{\text{ч}} = \frac{M}{T_m \cdot t} \text{ грн.},$$

де  $M$  – місячний оклад працівників;  $T_m$  – кількість робочих днів тиждень;  $t$  – кількість робочих годин в день.

$$C_{\text{ч}} = \frac{7000 + 7000 + 10000}{3 \cdot 21 \cdot 8} = 47.61 \text{ грн.}$$

Тоді, розрахуємо заробітну плату за формулою

$$C_{\text{ЗП}} = C_{\text{ч}} \cdot T_i \cdot K_{\text{д}},$$

де  $C_{\text{ч}}$  – величина погодинної оплати праці програміста;  $T_i$  – трудомісткість відповідного завдання;  $K_{\text{д}}$  – норматив, який враховує додаткову заробітну плату.

Зарплата розробників за варіантами становить:

$$\text{I. } C_{\text{ЗП}} = 47.61 \cdot 1195 \cdot 1.2 = 68272.74 \text{ грн.}$$

$$\text{II. } C_{\text{ЗП}} = 47.61 \cdot 1181.22 \cdot 1.2 = 67485.46 \text{ грн.}$$

Відрахування на єдиний соціальний внесок становить 22%:

$$\text{I. } C_{\text{Від}} = C_{\text{ЗП}} \cdot 0.22 = 68272.74 \cdot 0.22 = 15020.0 \text{ грн.}$$

$$\text{II. } C_{\text{Від}} = C_{\text{ЗП}} \cdot 0.22 = 67485.46 \cdot 0.22 = 14846.80 \text{ грн.}$$

Тепер визначимо витрати на оплату однієї машино-години ( $C_M$ ).

Так як одна ЕОМ обслуговує одного програміста з окладом 7000 грн., з коефіцієнтом зайнятості 0.2 то для однієї машини отримаємо:

$$C_{\Gamma} = 12 \cdot M \cdot K_3 = 12 \cdot 7000 \cdot 0.2 = 16800 \text{ грн.}$$

З урахуванням додаткової заробітної плати:

$$C_{ЗП} = C_{Г} \cdot (1 + K_3) = 16800 \cdot (1 + 0.2) = 20160 \text{ грн.}$$

Відрахування на єдиний соціальний внесок:

$$C_{ВІД} = C_{ЗП} \cdot 0.22 = 20160 \cdot 0.22 = 4435.2 \text{ грн.}$$

Амортизаційні відрахування розраховуємо при амортизації 25% та вартості ЕОМ – 8000 грн.

$$C_A = K_{ТМ} \cdot K_A \cdot Ц_{ПР} = 1.15 \cdot 0.25 \cdot 8000 = 2300 \text{ грн.,}$$

де  $K_{ТМ}$  – коефіцієнт, який враховує витрати на транспортування та монтаж приладу у користувача;  $K_A$  – річна норма амортизації;  $Ц_{ПР}$  – договірна ціна приладу.

Витрати на ремонт та профілактику розраховуємо як:

$$C_P = K_{ТМ} \cdot Ц_{ПР} \cdot K_P = 1.15 \cdot 8000 \cdot 0.05 = 460 \text{ грн.,}$$

де  $K_P$  – відсоток витрат на поточні ремонти.

Ефективний годинний фонд часу ПК за рік розраховуємо за формулою:

$$T_{ЕФ} = (D_K - D_B - D_C - D_P) \cdot t_z \cdot K_B = (365 - 104 - 8 - 16) \cdot 8 \cdot 0.9 = 1706.4$$

годин,

де  $D_K$  – календарна кількість днів у році;  $D_B$ ,  $D_C$  – відповідно кількість вихідних та святкових днів;  $D_P$  – кількість днів планових ремонтів устаткування;  $t_z$  – кількість робочих годин в день;  $K_B$  – коефіцієнт використання приладу у часі протягом зміни.

Витрати на оплату електроенергії розраховуємо за формулою:

$$C_{ЕЛ} = T_{ЕФ} \cdot N_C \cdot K_3 \cdot Ц_{ЕН} = 1706.4 \cdot 0.156 \cdot 0.2 \cdot 2.0218 = 107.64 \text{ грн.,}$$

де  $N_C$  – середньо-споживча потужність приладу;  $K_3$  – коефіцієнтом зайнятості приладу;  $Ц_{ЕН}$  – тариф за 1 кВт-годин електроенергії.

Накладні витрати розраховуємо за формулою:

$$C_H = Ц_{ПР} \cdot 0.67 = 8000 \cdot 0.67 = 5360 \text{ грн.}$$

Тоді, річні експлуатаційні витрати будуть:

$$C_{ЕКС} = C_{ЗП} + C_{ВІД} + C_A + C_P + C_{ЕЛ} + C_H$$

$$C_{ЕКС} = 17280 + 3801.6 + 2300 + 460 + 107.64 + 5360 = 29309.24 \text{ грн.}$$

Собівартість однієї машино-години ЕОМ дорівнюватиме:

$$C_{M-Г} = C_{EКС} / T_{EФ} = 29309.24 / 1706.4 = 17.18 \text{ грн/час.}$$

Оскільки в даному випадку всі роботи, які пов'язані з розробкою програмного продукту ведуться на ЕОМ, витрати на оплату машинного часу, в залежності від обраного варіанта реалізації, складає:

$$C_M = C_{M-Г} \cdot T$$

$$\text{I. } C_M = 17.18 \cdot 1190 = 20444.2 \text{ грн.};$$

$$\text{II. } C_M = 17.18 \cdot 1173.12 = 20154.2 \text{ грн.};$$

Накладні витрати складають 67% від заробітної плати:

$$C_H = C_{ЗП} \cdot 0.67$$

$$\text{I. } C_H = 68272.74 \cdot 0.67 = 45742,73 \text{ грн.};$$

$$\text{II. } C_H = 67485.46 \cdot 0.67 = 45215.25 \text{ грн.};$$

Отже, вартість розробки ПП за варіантами становить:

$$C_{ПП} = C_{ЗП} + C_{Вид} + C_M + C_H$$

$$\text{I. } C_{ПП} = 68272.74 + 15020.0 + 20444.2 + 45742,73 = 149479,67 \text{ грн.};$$

$$\text{II. } C_{ПП} = 67485.46 + 14846.80 + 20154.2 + 45215.25 = 147701.71 \text{ грн.};$$

## **4.6 ВИБІР КРАЩОГО ВАРІАНТА ПП ТЕХНІКО-ЕКОНОМІЧНОГО РІВНЯ**

Розрахуємо коефіцієнт техніко-економічного рівня за формулою:

$$K_{TEPj} = K_{Kj} / C_{Фj},$$

$$K_{TEP1} = 2.879 / 132908.2 = 0.22 \cdot 10^{-4};$$

$$K_{TEP2} = 3.803 / 131022.9 = 0.29 \cdot 10^{-4};$$

Як бачимо, найбільш ефективним є другий варіант реалізації програми з коефіцієнтом техніко-економічного рівня  $K_{TEP1} = 0.29 \cdot 10^{-4}$ .

## 4.7 ВИСНОВОК

В даному розділі проведено повний функціонально-вартісний аналіз ПП, який було розроблено в рамках дипломного проекту. Процес аналізу можна умовно розділити на дві частини.

В першій з них проведено дослідження ПП з технічної точки зору: було визначено основні функції ПП та сформовано множину варіантів їх реалізації; на основі обчислених значень параметрів, а також експертних оцінок їх важливості було обчислено коефіцієнт технічного рівня, який і дав змогу визначити оптимальну з технічної точки зору альтернативу реалізації функцій ПП.

Другу частину ФВА присвячено вибору із альтернативних варіантів реалізації найбільш економічно обґрунтованого. Порівняння запропонованих варіантів реалізації в рамках даної частини виконувалось за коефіцієнтом ефективності, для обчислення якого були обчислені такі допоміжні параметри, як трудомісткість, витрати на заробітну плату, накладні витрати.

Після виконання функціонально-вартісного аналізу програмного комплексу що розроблюється, можна зробити висновок, що з альтернатив, що залишились після першого відбору двох варіантів виконання програмного комплексу оптимальним є перший варіант реалізації програмного продукту. У нього виявився найкращий показник техніко-економічного рівня якості  $K_{\text{TEP}} = 0.29 \cdot 10^{-4}$ .

Цей варіант реалізації програмного продукту має такі параметри:

- ✓ середовище проектування NetBeans;
- ✓ створення своєї онтології;
- ✓ Знаходження найоптимальніших алгоритмів навчання для студентів.

## ВИСНОВКИ

Результатом досліджень адаптивних технологій контролю знань було запропоновано ввести модель викладача. Головна її задача поєднати в одне ціле навчальну інформацію, що дається викладачем на лекціях і питання та завдання, що формують перевірку знань студентів. Введення нової моделі змінило модель адаптивного контролю знань, адже вплив першої поширюється майже на всі компоненти останньої. Такі зміни значно покращать уявлення про систему контролю знань успішності студентів.

Інтеграція побудови інформації з навчального предмету дозволяє структурувати навчальний матеріал; виконати досить повне його подання; переривати контроль знань залежно від умов, що визначаються викладачем; мінімізувати інформаційну надмірність і час тестування.

Запропонована система генерації питань. Показано можливості використання її для складання тестових питань закритого типу, які дозволяють отримувати завдання достатньо високої складності та носять педагогічну цінність.

Обґрунтовано актуальність досліджень в сфері автоматичної генерації тестових завдань, яка спрямована на економію людських ресурсів, проаналізовано основні відомі підходи, визначено основні напрями розвитку даної проблематики.

Побудовано алгоритм адаптивного контролю знань, який включає як тематичні тестування (тобто окремий тест для кожної теми), так і підсумкове. Ця система зосереджена тільки на тестовому контролі знань. Звичайно, не можна виставити оцінку, опираючись на дані тестування, але слід обов'язково її враховувати.



Результатом цих всіх досліджень було створено прототип інтегрованої системи навчання і контролю знань на основі адаптивних технологій. Можливості цієї системи показало, що дана технологія є однією з найкращих для підготовки висококваліфікованих спеціалістів. Ця система дозволяє знайти індивідуальний підхід до кожного студента, що з одного боку полегшує освоєння матеріалу, а з іншого скорочує людські ресурси викладача.

Звичайно не варто спиратися лише на результати тестів для виставлення оцінки студенту, оскільки система відносно молода і потребує до опрацювань, але її результати багато чого можуть сказати про рівень знань студентів, що тим самим полегшує оцінювання студента.

Адаптивне навчання дозволяє навчати без звичних, для масової освіти, численних прогалин в індивідуальній підготовленості учнів та студентів, досягати необхідної структури та бажаного рівня знань. Останні тенденції в освітніх процесах показують актуалізацію технологій адаптивного навчання та тестування та необхідність змін в класичному підході до навчання.

## ПЕРЕЛІК ПОСИЛАНЬ

1. Прокофьева Н. О. Модели и методы компьютерного контроля знаний. [Электронный ресурс] – Режим доступа: [http://alephfiles.rtu.lv/TUA01/000029477\\_e.pdf](http://alephfiles.rtu.lv/TUA01/000029477_e.pdf) (дата звернення: 13.02.2016 )
131. Зайцева Л.В. Некоторые аспекты контроля знаний в дистанционном обучении // Образование и виртуальность - 2000. Сборник научных трудов 4-й Международной конференции. - Харьков - Севастополь: УАДО, 2000, - с.126-131
3. Зайцева Л.В. Проблемы компьютерного контроля знаний / Зайцева Л.В., Прокофьева Н.О. // Proceedings. IEEE International Conference on Advanced Learning Technologies (ICALT 2002). 9-12 September 2002. Kazan, Tatrstan, Russia, 2002, - p. 102 - 106.
4. Растрингин Л.А. Адаптивное обучение с моделью обучаемого. / Растрингин Л.А., Эренштейн М.Х. // РПИ. - Рига : Зинатне, 1986. - 160 с.
5. Зайцева Л.В. Модели и методы адаптивного контроля знаний / Зайцева Л.В., Прокофьева Н.О. // Educational Technology & Society. - Nr.7(4), 2004 ISSN 1436-4522 [Электронный ресурс] – Режим доступа: <http://ifets.ieee.org/russian/periodical/journal.html> (дата звернення: 25.02.2016 )
6. Буль Е.Е. Обзор моделей студента для компьютерных систем обучения // Educational Technology & Society 6(4). – 2003. ISSN 1436-4522
7. Коляда М.Г. Виды моделей обучаемых в автоматизированных обучающих системах // Искусственный интеллект. – 2008. – № 3. – С. 142–147
8. Мелещенко Т.В. Психолого-педагогічна модель викладача вищої школи та шляхи її реалізації // Вісник психології і педагогіки. – 2010. [Электронный ресурс] – Режим доступа: <http://www.psych.kiev.ua> (дата звернення: 03.03.2016 )
9. Генсьорська М. М. Адаптивне тестування в освіті // Інформаційно-комунікаційні технології в освіті. – 2014. - №1 [Электронный ресурс] – Режим

доступу: <http://www.e-journals.npu.edu.ua/index.php/ikt/article/viewFile/36/pdf20>

(дата звернення: 22.04.2016 )

- 10.** Weiss D. J.(Ed.) *New Horizons in Testing: Latent Trait. Test Theory and Computerised Adaptive Testing.* N.Y., Academic Press, 1983. – 345 pp.
- 11.** Lord P. M. *Application of Item Response Theory to Practical Testing Problems.* Hillsdale N – J. Lawrence Erlbaum Ass., Publ. 1980, – 266 pp.
- 12.** Норенков И. П. Интеллектуальные технологии на основе онтологий // Информационные технологии. – 2010. – № 1. – С. 17-23.
- 13.** Онтологии и тезаурусы: модели, инструменты, приложения / Добров Б.В., Иванов В.В., Лукашевич Н.В., Соловьев В.Д. – М.: БИНОМ, 2009. – 173 с.
- 14.** T. R. Gruber. A translation approach to portable ontologies. *Knowledge Acquisition*, 5(2):199-220, 1993.
- 15.** Онтології і подання знань [Електронний ресурс] – Режим доступу: <http://www.znannya.org/?view=ontology-give-knowledge> (дата звернення: 10.03.2016 )
- 16.** Манцивода А.В. Достижения в Интернете и будущее информационной среды российского образования / Манцивода А.В., Малых А.А. Информационные технологии. – 2008. – № 1. – С. 67-74.
- 17.** Никоненко А.А. Обзор баз знаний онтологического типа // Искусственный интеллект. – 2002. – № 4. – С. 157–163.
- 18.** Шатовская Т. Репозиторий интеллектуального анализа данных / Шатовская Т. Каменева И., Гуд А. // Компьютерные науки и информационные технологии. - 2009. - № 650. - С. 263-269.
- 19.** Клещев А. С. Системный анализ при автоматизации интеллектуальной профессиональной деятельности / Клещев А. С., Шалфеева Е. А. // Труды XIII национальной конф. по искусственному интеллекту КИИ-2012. – Т.2. – Белгород: БГТУ, 2012. – С. 128–135.
- 20.** Палагин А.В. Системно-онтологический анализ предметной области /

- Палагин А.В., Петренко Н. Г. // Управляющие системы и машины. - 2009. – № 4. - С. 3-14.
- 21.** Норенков И. П. Интеллектуальные технологии на основе онтологий // Информационные технологии. – 2010. – № 1. – С. 17-23.
- 22.** Снитюк В.Е. Интеллектуальное управление оцениванием знаний / Снитюк В.Е., Юрченко К.Н. – Черкассы, 2013. – 262 с.
- 23.** Noy N.F. Ontology Development 101: A Guide to Creating Your First Ontology / Noy N.F., McGuinness D.L. // Knowledge Systems Laboratory Technical Report KSL-01-05 and Medical Informatics Technical Report SMI-2001-0880. – Stanford. –2001.–23 p.
- 24.** Титенко С.В. Семантическая модель знаний для целей организации контроля знаний в обучающей системе / Титенко С.В., Гагарин О.О. // Сб. трудов VI Межд. конф. —Интеллектуальный анализ информации-2006. – К., 2006. – С. 298-307.
- 25.** Методика извлечения знаний при построении интеллектуальных обучающих систем / Таран Т.А., Копычко С.Н., Сирота С.В., Гулякина Н.А. // Сб. трудов VI Межд. конф. —Интеллектуальный анализ информации-2006. – К., 2006. – С. 282-287.
- 26.** Таран Т.А. Обучение понятиям в интеллектуальных обучающих системах на основе формального концептуального анализа / Таран Т.А., Сирота С.В. // Искусственный интеллект. – 2000. – № 3. – С. 340-347.
- 27.** Нетавская Е. Концептуальные принципы реализации и структура инструментария контроля знаний на базе онтологий // In Proc. XIIIth Int. Conf. —Knowledge-Dialogue-Solutions. – Bulgaria, Varna, 2007; Vol. 2. – P. 464-470.
- 28.** Гайтан О. М. Елементи технології реалізації автоматизованого адаптивного

контролю знань студентів в комп'ютерних системах навчання // ISSN 1814-4225. Радіоелектронні і комп'ютерні системи. – 2014. – № 4 (68)

**29.** Танченко С. С. Анализ методов генерации тестовых заданий / Танченко С.

С., Титенко С. В., Гагарин А. А. // XIII международная научная конференция имени Т. А. Таран «Интеллектуальный анализ информации ИАИ-2013», Киев, 15-17 мая 2013 г. : сб. тр./ гл. ред. С.В.Сирота. – К. : Просвіта, 2013. – С. 220-226

**30.** Мельник А. М. Метод генерації тестових завдань на основі системи семантичних класів / Мельник А.М., Пасічник Р.М. // Вісник ТДТУ. — 2010. —

Том 15. — № 1. — С. 187-193. Петрова Л.Г. Використання модифікованої понятійно-тезисної моделі для автоматизованого формування бази тестових запитань в системах комп'ютеризації освіти / Петрова Л.Г., Петров С.О. // Інформаційні технології і засоби навчання. — 2012. — №4 (30).

**31.** Рафальська О. О. Адаптивне тестування в системі Moodle. [Електронний ресурс] – Режим доступу: <http://2015.moodlemoot.in.ua/course/view.php?id=119&lang=ru> (дата звернення: 03.06.2016)

**32.** Правила охорони праці під час експлуатації електронно-обчислювальних машин : НПАОП 0.00.-1.31-10. – [Чинний від 2010-03-26]. – К. : Держнагляд охорони праці України, 2010. – 7 с. – (Національні стандарти України).

**34.** Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин ДСанПіН 3.3.2.007-98

(затверджено Постановою Головного державного санітарного лікаря України від 10.12.1998 р. № 7).

**35.** Санітарні норми виробничого шуму, ультразвуку та інфразвуку : ДСН

3.3.6.037-99. – [Чинний від 2000-01-01]. – К. : МОЗ України, 2000. – 37 с. – (Національні стандарти України).

**36.** Природне і штучне освітлення : ДБН В.2.5-28:2015 – [Чинний від 2015-01-

01]. – К. : Міністерство будівництва, архітектури та житлово-комунального господарства України, 2015. – 171 с. – (Національні стандарти України).

**37.** Охорона праці в офісі. Вимоги до робочого місця офісного працівника –

[Електронний ресурс] - Режим доступу: <http://gc.ua/business-news/oxorona-praci-v-ofisi-vimogi-do-robochogo-miscya-ofisnogo-pracivnika/> (дата звернення: 01.06.2016)

**38.** Санітарні норми мікроклімату виробничих приміщень : ДСН 3.3.6.042-99. – [Чинний від 2000-01-01]. – К. : МОЗ України, 2000. – 42 с. – (Національні стандарти України).

**39.** Норми визначення категорій приміщень, будинків та зовнішніх установок за вибухопожежною та пожежною небезпекою. НАПБ Б.03.002-2007. (затверджено наказом МНС України від 03.12.2007 № 833)