

ПОЛТАВСЬКИЙ УНІВЕРСИТЕТ ЕКОНОМІКИ І ТОРГІВЛІ

Навчально-науковий інститут денної освіти
Форма навчання денна
Кафедра комп'ютерних наук та інформаційних технологій

Допускається до захисту
Завідувач кафедри
_____ Олена ОЛЬХОВСЬКА
(підпис)

«___» _____ 202_ р.

КВАЛІФІКАЦІЙНА РОБОТА

на тему

**«НАВЧАЛЬНИЙ ТРЕНУВАЛЬНИЙ ІНСТРУМЕНТ ТА ЙОГО ПРОГРАМУВАННЯ
З ТЕМИ «КВАНТОРИ. ФОРМУЛИ ЛОГІКИ ПРЕДИКАТИВ» ДИСТАНЦІЙНОГО
КУРСУ «МАТЕМАТИЧНА ЛОГІКА»**

зі спеціальності 122 Комп'ютерні науки
освітня програма «Комп'ютерні науки»
ступеня магістра

Виконавець роботи Токар Михайло Олександрович

_____ «___» _____ 202_ р.
(підпис)

Науковий керівник к.ф.-м.н., доц. Парфьонова Тетяна Олександрівна

_____ «___» _____ 202_ р.
(підпис)

Рецензент _____

ПОЛТАВА 2023

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	3
ВСТУП.....	4
1. ПОСТАНОВКА ЗАВДАННЯ.....	6
2. ІНФОРМАЦІЙНИЙ ОГЛЯД	7
2.1. Переваги та недоліки дистанційного навчання	7
2.2. Основні види та призначення платформ дистанційного навчання.....	8
2.3. Необхідність та актуальність теми роботи.....	10
3. ТЕОРЕТИЧНА ЧАСТИНА.....	12
3.1 Основні поняття предиката, операції та побудова формул	12
3.2. Кванторні операції.....	14
3.3. Алгоритм роботи програми	17
3.4. Обґрунтування вибору мови програмування та середовища розробки	22
4. ПРАКТИЧНА ЧАСТИНА.....	24
4.1. Блок-схеми алгоритму.....	24
4.2. Опис програми	32
4.3. Інструкція користувачу	34
ВИСНОВКИ	41
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ.....	42
ДОДАТОК А. КОД ПРОГРАМИ.....	44

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

C#	об'єктно-орієнтована мова програмування з безпечною системою типізації для платформи .NET.
Microsoft Visual Studio	серія продуктів фірми Майкрософт, які включають інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів
Предикат	функція з областю значень $\{0,1\}$ (або «Істина» та «Хибна»), певна на n -й декартовій ступені множини M .
Логіка предикатів	Розділ класичної символічної логіки, що вивчає суб'єктно-предикатну структуру висловлювань

ВСТУП

У наш час, коли комп'ютерні технології та дистанційне навчання набувають великого розвитку, вони відкривають перед нами широкі можливості для модернізації та підвищення ефективності навчальних програм. Цей процес стає особливо значущим завдяки тренажерам.

Розширення дистанційного навчання призвело до з'яви різноманітних технологій, навчальних програм і систем, які збагачують та підвищують ефективність навчального процесу. Протягом останніх років, дистанційні курси стали не просто альтернативою, але головним інструментом навчання для багатьох. Це свідчить про значний розвиток і зростання популярності дистанційної освіти в сучасному світі.

Дистанційне навчання передбачає більше, ніж просто передачу інформації та знань. Воно сприяє:

- розвитку інтелектуальних здібностей через конструктивне та алгоритмічне мислення, які формуються завдяки взаємодії з комп'ютерами;
- розвитку творчого мислення завдяки зменшенню частки репродуктивної активності;
- професійному зростанню шляхом навчання приймати оптимальні професійні рішення у складних ситуаціях, таких як комп'ютерні ділові ігри та робота з програмами-тренажерами.

Дистанційне навчання стало важливим кроком у сучасному освітньому процесі, що допомагає зростити ефективність та доступність навчання, а також сприяє розвитку різних аспектів інтелекту, творчості та професійної компетентності.

Мета кваліфікаційної роботи – проєктування тренувального інструменту та його програмування з теми «Квантори. Формули логіки предикатів» дистанційного курсу «Математична логіка».

Об'єктом розробки в даній роботі є створення програмних засобів дистанційного вивчення дисципліни «Математична логіка».

Предмет розробки – програмний продукт, що реалізує тренажер для закріплення знань під час вивчення кванторів та формул логіки предикатів.

Головне завдання – алгоритмізація та програмна реалізація роботи тренажеру з теми «Квантори. Формули логіки предикатів» .

Методи розробки — методика застосування кванторів та формул логіки предикатів. Серед програмного забезпечення – середовище розробки Visual Studio, мова програмування C# із застосуванням Windows Form.

Кваліфікаційна робота складається з чотирьох розділів. У першому розділі розглянута постановка завдання. У другому розділі розібрані переваги та недоліки дистанційного навчання також основні види і призначення платформ дистанційного навчання та необхідність та актуальність теми роботи. У третьому розділі розглянуто основні поняття предиката, операції та побудова формул, теоретичні відомості по кванторам, алгоритм програми та обґрунтування вибору мови програмування та середовища розробки. У четвертому розділі блок-схеми алгоритму, опис програми та інструкція користувача. Обсяг пояснювальної записки: 55 стор., в т.ч. основна частина - 43 стор., джерела - 12 назв.

1. ПОСТАНОВКА ЗАВДАННЯ

Основні завдання роботи:

- розглянути переваги та недоліки дистанційного навчання;
- вказати основні види та призначення платформ дистанційного навчання;
- вказати актуальність теми;
- розглянути теоретичні відомості;
- розробити алгоритм тренажера, що дозволить закріпити знання і навички;
- обґрунтувати вибір програмних засобів;
- створити програмну реалізацію тренажера;
- зробити опис розробленої програми;
- написати інструкцію користувача.

Оскільки навчальний тренувальний інструмент створюється для дистанційного курсу, то потрібно описати переваги і недоліки дистанційного навчання. Під час розробки алгоритму слід передбачити можливість для користувача звернутися до теоретичного матеріалу, що стосується обраної теми. Це допоможе користувачу успішно виконати завдання і систематично описати всі кроки при вирішенні цього ж завдання.

Розроблений тренажер повинен мати зрозумілий для користувача інтерфейс.

Для цього він повинен містити наступні елементи:

1. головну сторінку;
2. можливість ознайомитися з теоретичним матеріалом;
3. виведення завдання;
4. варіанти відповідей до завдання;
5. виведення повідомлення про помилку;
6. Кінцеву сторінку із результатами проходження та можливістю зробити наступну спробу.

При програмній реалізації потрібно розробити виведення повідомлення про помилку у разі неправильної відповіді. Це повідомлення має містити правильну відповідь[1].

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Переваги та недоліки дистанційного навчання

Дистанційне навчання стало невід'ємною частиною життя будь-якої людини, звичайно ж воно має свої переваги і недоліки. З одного боку, воно надає:

- Гнучкість і доступність: Дистанційне навчання дозволяє будь-якій людині навчатися з будь-якого місця, забезпечуючи гнучкість графіку та доступ до освіти для різних груп людей.
- Самостійність: Воно сприяє розвитку самостійності та саморегуляції у людини, оскільки вони великою мірою відповідають за своє навчання.
- Великий вибір предметів і курсів: Кожна людина може вибирати з різноманіття курсів та програм, незалежно від їхнього місця проживання.
- Ефективне використання часу: Дистанційне навчання дозволяє оптимізувати використання свого часу, навчаючись в зручний для них час і темп.
- Індивідуальний підхід: При традиційному навчанні викладачеві досить важко приділити необхідну кількість уваги всім студентам групи, підлаштовуватися під темп роботи кожного. Використання дистанційних технологій підходить для організації індивідуального підходу. Крім того, що студент сам обирає собі темп навчання, він може оперативно отримати у викладача відповіді на виникаючі питання[2].

Окрім списку переваг, дистанційне навчання не позбавлене і ряду недоліків:

- Відсутність фізичного контакту: Дистанційне навчання може призвести до відчуття відокремленості від викладачів та однокурсників через відсутність фізичного контакту та спілкування.
- Вимога до самодисципліни: Для успішного дистанційного навчання потрібна висока самодисципліна та мотивація, і брак цих якостей може вплинути на результативність.

- Технічні проблеми: Для дистанційного навчання необхідний доступ до технологій та стабільного Інтернет-з'єднання, і технічні проблеми можуть вплинути на навчання.
- Відсутність спільноти: Дистанційне навчання може відірвати від соціальної спільноти та можливості взаємодії з однокурсниками та викладачами, що може вплинути на соціальний і академічний досвід.
- Проблема ідентифікації: Поки найефективніший спосіб простежити за тим, чи людина самотійно здавала іспити чи заліки, - це відеоспостереження, що не завжди можливо.

Звісно, вибір дистанційного навчання також пов'язаний з викликами і вимагає від людини високого рівня самодисципліни та внутрішньої мотивації. Але для багатьох людей незважаючи на недоліки, переваги роблять дистанційне навчання відмінним варіантом для отримання освіти[3].

2.2. Основні види та призначення платформ дистанційного навчання

Одним із стратегічних напрямків реформування системи освіти України є активне використання інформаційно-комунікаційних технологій для розвитку дистанційної освіти. Вибір платформи для дистанційного навчання є дуже важливим кроком.

Одним з лідерів платформ дистанційного навчання є HUMAN - українська ed-tech компанія, яка розробляє та впроваджує продукти для діджиталізації освітнього процесу в Україні з урахуванням матеріально-технічного забезпечення та цифрової грамотності навчальних закладів різних рівнів. HUMAN забезпечує комплексну діджиталізацію навчальних закладів, інтегруючи в єдину систему інструменти для організації, комунікації та аналізу освітніх процесів. В результаті заклади отримують прозору статистику освітніх процесів, а також інструменти для очного та дистанційного навчання.

Google Classroom - це безкоштовний веб-сервіс, створений компанією Google для навчальних закладів, щоб спростити створення, розподіл і категоризацію

завдань у без паперовий спосіб. Основна мета сервісу - прискорити процес обміну файлами між викладачами та студентами. Його можуть використовувати вчителі та учні в школах, а також викладачі та студенти у вищих навчальних закладах. Google Classroom поєднує в собі такі функції: Google Диск для створення та обміну завданнями; Google Документи, Таблиці та Слайди для написання документів; Gmail для спілкування; Google Календар для планування; та інші доступні інші веб-сервіси. Учні запрошуються на заняття за допомогою приватного коду або автоматично імпортуються зі шкільного веб-сайту. У кожному класі створюється окрема папка на Google Диску кожного користувача, куди учні подають свої роботи, які потім оцінюються вчителем. мобільні додатки, доступні для iOS та Android, дозволяють учням робити фотографії та прикріплювати їх до завдань, ділитися файлами з іншими учнями, а також обмінюватися файлами з інших Мобільний додаток, доступний для iOS та Android, дозволяє користувачам робити фотографії та прикріплювати їх до завдань, ділитися файлами з інших додатків, а також отримувати доступ до інформації в режимі офлайн. Вчителі можуть відстежувати прогрес кожного учня, оцінювати їхні роботи та повертати їх з коментарями.

Сайт "Нові знання" є однією зі складових всеукраїнського проекту "КУРС: Освіта". Користувачами є керівники (директори та їхні заступники), завучі, вчителі, учні та батьки загальноосвітніх навчальних закладів України.

Moodle (модульне об'єктно-орієнтоване динамічне навчальне середовище) - це навчальна платформа, призначена для об'єднання вчителів, адміністраторів та учнів в одну надійну, безпечну та інтегровану систему для створення персоналізованого навчального середовища. Вона реалізує філософію "соціальної конструктивістської педагогіки" і зосереджена насамперед на організації взаємодії вчителя та учня, але також підходить для організації традиційних дистанційних курсів та підтримки очного навчання Moodle - це платформа електронного навчання, система управління курсами (CMS), система управління навчанням (LMS) або віртуальне навчальне середовище (VLE).) або віртуальне навчальне середовище (VNC); Moodle дозволяє викладачам створювати ефективні

сайти для онлайн-навчання; Moodle можна використовувати не лише для навчання школярів та студентів, але й для професійного розвитку та бізнес-тренінгів.

Типові можливості Moodle включають

1. подання завдань
2. дискусійні форуми
3. завантаження файлів
4. оцінювання
5. обмін повідомленнями
6. календар подій
7. новини та анонси подій (за рівнями: сайт, курси, навчальні групи)
8. онлайн-тести
9. вікі

За замовчуванням Moodle включає бібліотеку TCPDF, яка може генерувати документи з PDF сторінок[4].

2.3. Необхідність та актуальність теми роботи

Інформатизація суспільства – це глобальний соціальний процес, особливість якого полягає в тому, що домінуючим видом діяльності в сфері суспільного виробництва є збір, накопичення, продукування, обробка, зберігання, передача та використання інформації, здійснювані на основі сучасних засобів мікропроцесорної та обчислювальної техніки, а також на базі різноманітних засобів інформаційного обміну.

Досить часто дистанційне навчання плутають із заочною освітою, але це не одне і теж. Якщо, навчаючись на заочному, студент зустрічається з викладачем на вичитування лекцій та іспитах, а основний матеріал освоює самостійно і до всього доходить сам, то дистанційне навчання має на увазі, що обидві сторони знаходяться на зв'язку практично постійно і спілкуються за допомогою передових технологій.

Головною метою створення системи дистанційної освіти є забезпечення загальнонаціонального доступу до освітніх ресурсів шляхом використання сучасних

інформаційних технологій та телекомунікаційних мереж і створення умов для реалізації громадянами своїх прав на освіту.

Гнучкість дистанційного навчання полягає у можливості викладання матеріалу курсу з урахуванням підготовки та здібностей студентів. Це досягається створенням альтернативних сайтів для одержання більш детальної або додаткової інформації із складних тем, або низки питань підказок.

Вивчення досвіду впровадження дистанційної форми навчання в системи освіти різних країн дозволило виділити основні мотиваційні причини, що спонукають навчальні заклади до організації системи дистанційного навчання: поліпшення якості навчання; переваги нових педагогічних технологій; зростаючий попит на нову форму навчання; отримання доходів; можливість постійної взаємодії викладачів і студентів; необхідність виживання в інформаційному суспільстві; можливість скорочення витрат на реорганізацію освіти[5].

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1 Основні поняття предиката, операції та побудова формул

Логіка предикатів — це розділ класичної символічної логіки, що вивчає суб'єктно-предикатну структуру висловлювань, на підставі чого визначають значення істинності висловлювань; по-іншому — це дедуктивна теорія, яка моделює процес виведення одних висловлювань із інших, враховуючи їх структуру. Логіку предикатів трактують як розширення логіки висловлювань через виявлення внутрішньої структури висловлювань і введення нових термінів та системи аксіом.

Логіка предикатів як система створюється відповідно до загальних принципів побудови формальних систем. Особливість логіки предикатів полягає в тому, що вона є складнішою і за семантикою, і за синтаксисом порівняно з логікою висловлювань. Розрізняють семантику та синтаксис логіки предикатів.

У семантичному аспекті визначають суб'єктно-предикатну структуру висловлювань на змістовному рівні. Це дає змогу виявити властивості, притаманні певній сукупності емпіричних або абстрактних об'єктів, і ввести терміни, котрі відокремлюють сферу дії предикатів: висловлювання, властивість, відношення, предикат, одномісний предикат, багатомісний предикат, квантор загальності, квантор існування, істинне значення висловлення.

Висловлення, в якому емпіричному чи абстрактному об'єктові приписують певну властивість P або визначаються відношення між об'єктами, надають два значення істинності: «істина» (i); «хиба» (x). Відповідно, логіка предикатів — двозначна за кількістю значень істинності висловлювань.

У синтаксичному аспекті суб'єктно-предикатну структуру висловлювань визначають у процесі абстрагування від їх змісту та формалізують засобами штучно створеної мови, на підставі чого здійснюють логічні операції над символами, що зображають ці відношення (числення предикатів).

Структура логіки предикатів — алфавіт, правила побудови формул із символів алфавіту, правила дедуктивного виведення з аксіом нових формул (доведення теорем), правила інтерпретації.

Мова логіки предикатів — це система символів, що створюють алфавіт. До нього належать символи, введені в логіці висловлювань, і нові символи, які позначають терміни, введені в логіці предикатів.

Предикат - це вираз або функція, яка виражає відношення між об'єктами або описує властивість об'єкта. В математичній логіці, предикат приймає аргументи (зазвичай змінні) і повертає значення істинності (істинний або хибний) в залежності від того, чи відповідають аргументи визначеному властивості.

Наприклад, предикат " $x > 5$ " визначає властивість, за якої число " x " більше 5. Якщо " x " дорівнює 7, то предикат " $x > 5$ " є істинним, оскільки 7 більше за 5. Але якщо " x " дорівнює 3, то предикат буде хибним, оскільки 3 менше за 5.

Предикати важливі для формалізації та виразування логічних відношень та властивостей в математиці та логіці. Вони використовуються для побудови логічних виразів, формулювання математичних тверджень та вирішення різних завдань, пов'язаних із виразами істинності або хибності в різних областях науки та інженерії.

Логічною системою є логіка предикатів, що містить логіку висловлень як свою складову частину. Логіка предикатів, як і традиційна формальна логіка, поділяє елементарне висловлення на суб'єкт і предикат.

Суб'єкт — це те, про що дещо стверджується у висловленні.

Предикат — це те, що стверджується про суб'єкт.

Наприклад, у висловленні “13 — просте число”, “13” — суб'єкт, “просте число” — предикат. Це висловлення стверджує, що “13” має властивість “бути простим числом”[6].

Побудова формул логіки предикатів

- Окремо взятий предикат називається елементарною формулою.
- Якщо F і Q — формули логіки предикатів, то $\neg F$, $(F \wedge Q)$, $(F \vee Q)$, $(F \rightarrow Q)$ — формули.
- $P(x)$ — формула, що виражає властивість (одномісний предикат).
- $R(x, y)$ — формула, яка виражає двомісний предикат.
- $R(x, y, z)$ — формула, що виражає тримісний предикат.
- Якщо P — формула і x — предметна змінна, то $\forall x P(x)$ і $\exists x P(x)$ є формулами.

Логічні операції над предикатами

- Запереченням.
- Кон'юнкцією.
- Диз'юнкцією.
- Імплікацією.
- Еквіваленцією.

3.2. Кванторні операції

Кванторні операції

Квантор (від лат. *quantum* – скільки)- це логічний оператор, який описує відношення між суб'єктом та предикатом і несе кількісну характеристику виразу, перед яким він поставлений.

Квантор загальності - це твердження, яке приписує властивість P деякому непорожньому класу в цілому, тобто означає, що властивість P є унікальною для всіх елементів класу A . Цей квантор має вираз "для всіх" ("all", "each", "any", "whatever"). Він позначається символом $\forall$, а повний вираз має вигляд $\forall x P(x)$ (тобто кожен x має властивість P). Наприклад, твердження "Всі люди в класі людей мають властивість бути смертними" представлено виразом $\forall x P(x)$.

Квантори існування - це твердження про деякий непорожній клас, для якого властивість P притаманна лише деяким елементам цього класу, тобто існують елементи класу A з властивістю P .

Екзистенціальний квантор має вираз "існує" ("деякий", "тільки один"). Він позначається символом $\exists$, а повний вираз має вигляд $\exists x P(x)$ (тобто "існує" x з властивістю P). Наприклад, речення "Деякі люди мають властивість писати вірші" ("Деякі люди пишуть вірші") представлено формулою $\exists x P(x)$.

Оскільки загальність і квантори існування взаємозалежні, всі логічні операції виконуються шляхом визначення логічних відношень між ними.

Чисельні квантори

У математиці часто зустрічаються вирази вигляду “щонайменше n ” (“хоча б n ”), “не більш ніж n ”, “ n і тільки n ” (“рівно n ”), де n – натуральне число.

Ці вирази, що називаються чисельними кванторами, мають чисто логічний зміст; вони можуть бути замінені рівнозначними виразами, що не містять числівників, складається тільки з логічних термінів, знаків $\equiv$ або $\sim$, що означає тотожність (збіг) об'єктів.

Нехай $n = 1$. Пропозиція “Щонайменше один об'єкт має властивість P ” має той же зміст, що і пропозиція “Існує об'єкт, що володіє властивістю P ”, тобто $\exists x(P(x))$. (*)

Пропозиція “не більш ніж один об'єкт має властивість P ” рівнозначна пропозиції “Якщо є об'єкти, що володіють властивістю P , то вони збігаються”, тобто $\forall x \forall y (P(x) \wedge P(y) \rightarrow x \equiv y)$. (**). Пропозицію “один і тільки один об'єкт має властивість P ” рівнозначно кон'юнкції вищевказаних пропозицій (*) і (**).

Заперечення пропозицій із кванторами

Відомо, що часто для заперечення деякої пропозиції досить подати присудкові цієї пропозиції негативну частку “не”. Наприклад, запереченням пропозиції “Ріка x впадає в Чорне море” є пропозиція “Ріка x не впадає в Чорне море”. Чи можливе використання цього прийому для побудови заперечень пропозицій із кванторами? Розглянемо приклад.

Пропозиції “Усі птахи літають” і “Усі птахи не літають” не є запереченнями одна одної, тому що вони обоє помилкові. Пропозиції “Деякі птахи літають” і “Деякі птахи не літають” не є запереченням один одного, тому що вони обоє істинні. Таким чином, пропозиції, що отримані додаванням частки “не” до присудка пропозицій “Усі x суть P ” і “Деякі x суть P ” не є запереченнями цих пропозицій. Універсальним способом побудови заперечення даної пропозиції є додавання словосполучення “невірно, що” на початку пропозиції. Таким чином, запереченням пропозиції “Усі птахи літають” є пропозиція “Невірно, що усі птахи літають”; але ця пропозиція має той же зміст, що і пропозиція “Деякі птахи не літають”. Запереченням пропозиції “Деякі птахи літають” є пропозиція “Невірно, що деякі птахи літають”, що має той же зміст, що і пропозиція “Усі птахи не літають”.

Домовимося заперечення пропозиції $\forall x(P(x))$ записувати як $\bar{\forall}x(P(x))$, а заперечення пропозиції $\exists x(P(x))$ – як $\bar{\exists}x(P(x))$. Очевидно, що пропозиція $\bar{\forall}x(P(x))$

має той же зміст, а отже, те ж значення істинності, що і пропозиція $\exists x(\overline{P(x)})$, а пропозиція $\exists x(P(x))$ – той же зміст, що $\forall x(\overline{P(x)})$. Інакше кажучи, $\forall x(P(x))$ рівносильно $\exists x(\overline{P(x)})$ ($\exists x(P(x))$ рівносильно $\forall x(\overline{P(x)})$).

Квантори загальності та існування називають двоїстими відносно один одного. З'ясуємо тепер, як будувати заперечення пропозиції, що починається з декількох кванторів, наприклад, такого: $\forall x \exists y \forall z (P(x, y, z))$.

Послідовно застосовуючи сформульоване вище правило, одержимо: $\forall x \exists y \forall z (P(x, y, z))$ рівносильно $\exists x (\exists y \forall z (P(x, y, z)))$, що рівносильно $\exists x \forall y (\forall z (P(x, y, z)))$, що рівносильно $\exists x \forall y \exists z (\overline{P(x, y, z)})$.

Область дії квантора

Область дії квантора визначає межі, в яких цей квантор діє. Вона відокремлюється дужками, що розташовані ліворуч і праворуч від виразу. Ліва дужка визначає початок сфери дії, а права дужка - закінчення цієї сфери. В межах цієї області виділяються два типи змінних: зв'язані та вільні.

Зв'язані змінні - це ті змінні, які входять до сфери дії кванторів загальності ($\forall$) або існування ($\exists$) або до обох одразу. Наприклад, у формулах типу $\forall x P(x)$ або $\exists x P(x)$, змінна x є зв'язаною, оскільки її значення обмежено самим квантором.

Вільні змінні - це змінні, які входять в певну формулу, але не входять в сферу дії кванторів загальності чи існування. Наприклад, у формулі типу $\forall (P(x)) \rightarrow Q(x)$, змінна x є вільною у виразі $Q(x)$, оскільки вона не обмежена квантором.

Логіку кванторів загальності ($\forall$) та існування ($\exists$) можна розглядати як узагальнення операцій кон'юнкції та диз'юнкції, коли множина значень змінної x є скінченною. Наприклад, якщо множина M складається зі значень x_1, x_2, x_3, x_4 , то формули типу $\forall x P(x)$ еквівалентні кон'юнкції $P(x_1) \wedge P(x_2) \wedge P(x_3) \wedge P(x_4)$, а формули типу $\exists x P(x)$ еквівалентні диз'юнкції $P(x_1) \vee P(x_2) \vee P(x_3) \vee P(x_4)$.

Отже, квантори загальності та існування дозволяють визначити, в якому обсязі висловлювання є істинними, і вони використовуються для встановлення логічних зв'язків між висловлюваннями.

3.3. Алгоритм роботи програми

Розглянемо роботу програми. При увімкненні програми бачимо головний екран, з якого можна перейти до теоретичної інформації, розпочати тестування або практичну частину.

Алгоритм роботи елементів теоретичної частини.

Користувач який невпевнений в своїх знаннях може переглянути теоретичну інформацію, яка допоможе з тестуванням. Для переходу використовується кнопка «Теоретична інформація» яка відкриває PDF файл з потрібною інформацією, в файлі знаходиться вся необхідна інформація для її повторення та проходження тестування.

Алгоритм роботи елементів тестування.

Для переходу до тесту використовується кнопка «Тестування» яка відкриває нову форму з питанням у вигляді тесту, який складається з 14 питань. Передбачено перемішування варіантів відповідей. Користувач повинен обрати одну або декілька з відповідей на екрані. Якщо було надано правильну відповідь, то тест переходить до наступного питання. Якщо обраний варіант є невірним програма повідомляє про помилку та підказує на правильну відповідь.

1. Квантор – це

A) логічна та бітова операція, що набуває значення «істина» тоді й лише тоді, коли значення «істина» має тільки один з її операндів

B) спеціальні символи для позначень вільних та зв'язаних змінних

C) логічний оператор, який описує відношення між суб'єктом та предикатом і несе кількісну характеристику виразу, перед яким він поставлений

D) спеціальний символ, який повідомляє транслятору про те, що потрібно виконати операцію з деякими двома операндами

2. Областю дії квантора називається

A) множина істинних значень формули логіки предикатів

B) множина хибних значень формули логіки предикатів

C) предикатні константи

D) предикат, на який даний квантор навішується

3. Які з представлених символів позначають кванторні операції? (Може бути декілька відповідей).

- $\wedge$
- $\exists$
- Σ
- $\in$
- $\Rightarrow$
- $\leftrightarrow$
- $\forall$
- $\oplus$

Правильна відповідь – $\exists$ та $\forall$.

4. Як позначається у логіці предикатів квантор загальності?

- A) $\exists$
- B) $\forall$**
- C) Σ
- D) $\in$

5. Вибрати відповідні слова, якими можна прочитати символ $\forall$ (Може бути декілька відповідей).

- **будь-який**
- існує
- деякий
- **кожен**
- **всі**
- хоч би один
- щонайменше один
- чи знайдеться

6. Яке твердження відповідає заданій формулі логіки предикатів:

$$\forall x(P(x) \rightarrow Q(x))?$$

A) існує x , для якого $P(x) \rightarrow Q(x)$ істинно

B) для всіх x , які задовольняють $P(x)$, також виконується $Q(x)$

C) існує x , для якого $P(x)$ і $Q(x)$ істинно

D) для кожного x , $P(x)$ виконується тоді і тільки тоді, коли справджується $Q(x)$

7. Яке твердження відповідає заданій формулі логіки предикатів

$$\exists y(P(y) \wedge Q(y))?$$

A) існує y , для якого істинні і $P(y)$ і $Q(y)$

B) для всіх y виконується або $P(y)$, або $Q(y)$

C) немає y , для якого одночасно виконується $P(y)$ і $Q(y)$

D) існує y , для якого $P(y)$ виконується тоді і тільки тоді, коли справджується $Q(x)$

8. Як позначається у логіці предикатів квантор існування?

A) $\exists$

B) $\forall$

C) $\in$

D) Σ

9. Вибрати відповідні слова, якими можна прочитати символ $\exists$

- будь-який
- **існує**
- деякий
- кожен
- всі
- **хоч би один**
- **щонайменше один**
- чи знайдеться

10. Вибрати змінні, які у формулі $\forall x \exists z (P(x) \rightarrow Q(z)) \wedge S(y)$ є зв'язаними

- x
- y
- z
- w

Правильна відповідь – x та z .

11. Вибрати змінні, які у формулі $\forall x (P(x) \leftrightarrow Q(y)) \rightarrow S(w)$ є вільними

- x
- y
- z
- w

Правильна відповідь – y та w .

12. Вибрати формулу логіки предикатів, яка відповідає висловленню: "Для всіх x одночасно виконуються $P(x)$ і $Q(x)$ "?

- A) $\forall x (P(x) \wedge Q(x))$
- B) $\exists x (P(x) \wedge Q(x))$
- C) $\exists x (P(x) \vee Q(x))$
- D) $\forall x (P(x) \leftrightarrow Q(x))$

13. Вибрати формулу логіки предикатів, яка відповідає висловленню: "Існує x , для якого $P(x)$ і $Q(x)$ істинне"?

- A) $\forall x (P(x) \wedge Q(x))$
- B) $\exists x (P(x) \wedge Q(x))$**
- C) $\exists x (P(x) \vee Q(x))$
- D) $\forall x (P(x) \rightarrow Q(x))$

14. Вибрати формулу логіки предикатів, яка відповідає висловленню: "Для всіх x виконується $P(x)$ або $Q(x)$ "?

A) $\forall x(P(x) \vee Q(x))$

B) $\exists x(P(x) \vee Q(x))$

C) $\forall x(P(x) \wedge Q(x))$

D) $\exists x(P(x) \leftrightarrow Q(x))$

Передбачено перемішування варіантів відповідей. Користувач повинен обрати одну з відповідей на екрані. Якщо було надано правильну відповідь, то тест переходить до наступного питання. Якщо обраний варіант є невірним програма повідомляє про помилку та підказує на правильну відповідь.

Після закінчення тесту програма надає результати скільки правильних та неправильних відповідей.

Тепер перед користувачем є вибір, де можна почати спочатку або повернутись на голову сторінку та перейти до практичної частини.

Алгоритм роботи елементів практичної частини.

На екрані представлено завдання практичної частини.

Поставити у відповідність твердженням формули, вважаючи що $S(x): x$ – студент, $P(x): x$ – склав іспит.

Всі студенти склали іспит	☐	1) $\forall x(S(x) \rightarrow P(x))$
Ніхто із студентів не склав іспит	☐	2) $\forall x(S(x) \rightarrow \overline{P(x)})$
Деякі студенти склали іспит	☐	3) $\exists x (S(x) \wedge P(x))$
Деякі студенти не склали іспит	☐	4) $\exists x (S(x) \wedge \overline{P(x)})$

Передбачено перемішування тверджень та формул, представлення їх випадковим чином.

Користувач має обрати відповідні числа в комірках. Якщо відповідь введено правильно, то програма показує результати. Якщо помилився – підказка у вигляді правильних відповідей та перехід до головного екрану де можна спробувати знову.

3.4. Обґрунтування вибору мови програмування та середовища розробки

Вибір Visual Studio та мови програмування C# надає багато важливих переваг та можливостей, які роблять цей інструментарій найкращим вибором для розробки програмного забезпечення. Нижче наведено основні причини, чому обрано саме Visual Studio та C#[7]:

1. Інтегроване середовище розробки (IDE): Visual Studio надає потужне та інтуїтивно зрозуміле інтегроване середовище розробки, яке полегшує роботу над проектами, редагування коду, налагодження та виправлення помилок. Немає необхідності встановлювати окремі інструменти, все знаходиться під одним дахом.

2. C# та Windows Form: C# є високорівневою мовою програмування, яка надає зручний синтаксис та велику кількість готових бібліотек. Windows Form – це технологія для створення графічного інтерфейсу користувача, що дозволяє швидко розробляти інтерактивні додатки.

3. Мова програмування C# - це сучасна об'єктно-орієнтована мова програмування зі строгою статичною типізацією, яка дозволяє створювати надійне та ефективне програмне забезпечення. Вона має багато функцій, які спрощують розробку, наприклад, автоматичне керування пам'яттю та збір сміття.

4. Широкі можливості розробки: використовуючи Visual Studio і C#, ви можете розробляти безліч різних типів додатків, включаючи консольні додатки, додатки з графічним інтерфейсом, веб-додатки, веб-сервіси і мобільні додатки для різних платформ. Вибір цих інструментів робить їх універсальними для багатьох завдань розробки[8].

5. Багатомовна підтримка: хоча Visual Studio підтримує кілька мов програмування, вона спеціально розроблена для роботи з C#. Це дозволяє використовувати інші мови в міру необхідності, зберігаючи при цьому зручність роботи на основній мові.

6. Велика спільнота та підтримка: Visual Studio та C# мають великі та активні спільноти розробників, де можна знайти відповіді на питання, приклади коду та корисні ресурси. Microsoft також надає активну підтримку та оновлення.

7. Незалежність від платформи: хоча Visual Studio та C# розроблені Microsoft, їх можна використовувати для розробки додатків для різних платформ, включаючи Windows, Android та iOS[9].

8. Безкоштовні версії для стартапів та студентів Microsoft пропонує безкоштовні версії Visual Studio Community для студентів, стартапів та розробників з відкритим вихідним кодом, щоб зменшити витрати на інструменти розробки.

9. C# та Windows Form: Visual Studio надає високий рівень інтеграції для налагодження коду та виконання тестів, що полегшує виявлення та виправлення помилок[10].

Загалом, Visual Studio та C# - це потужні інструменти для розробки програмного забезпечення з широким спектром можливостей та підтримки, що робить їх чудовим вибором для багатьох проектів розробки

4. ПРАКТИЧНА ЧАСТИНА

4.1. Блок-схеми алгоритму

На зображеннях 4.1.1 – 4.1.16 зображена блок-схема алгоритму роботи програми-тренажера.



Рисунок 4.1.1 – Блок-схема теоретичної частини

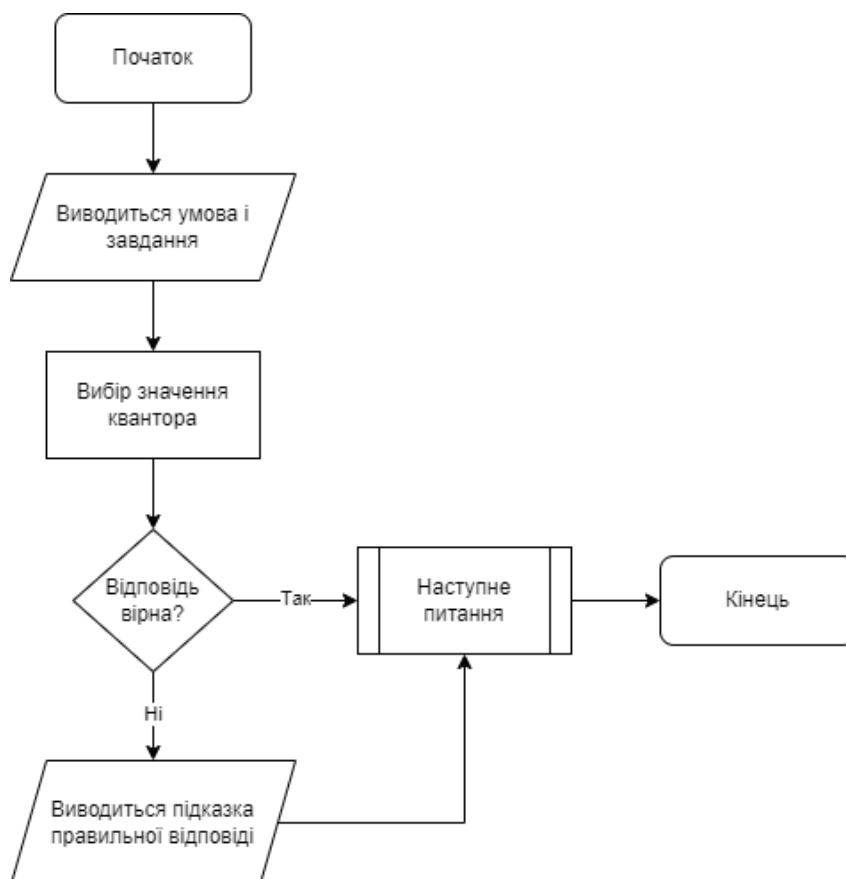


Рисунок 4.1.2 – Блок-схема першого питання тестування

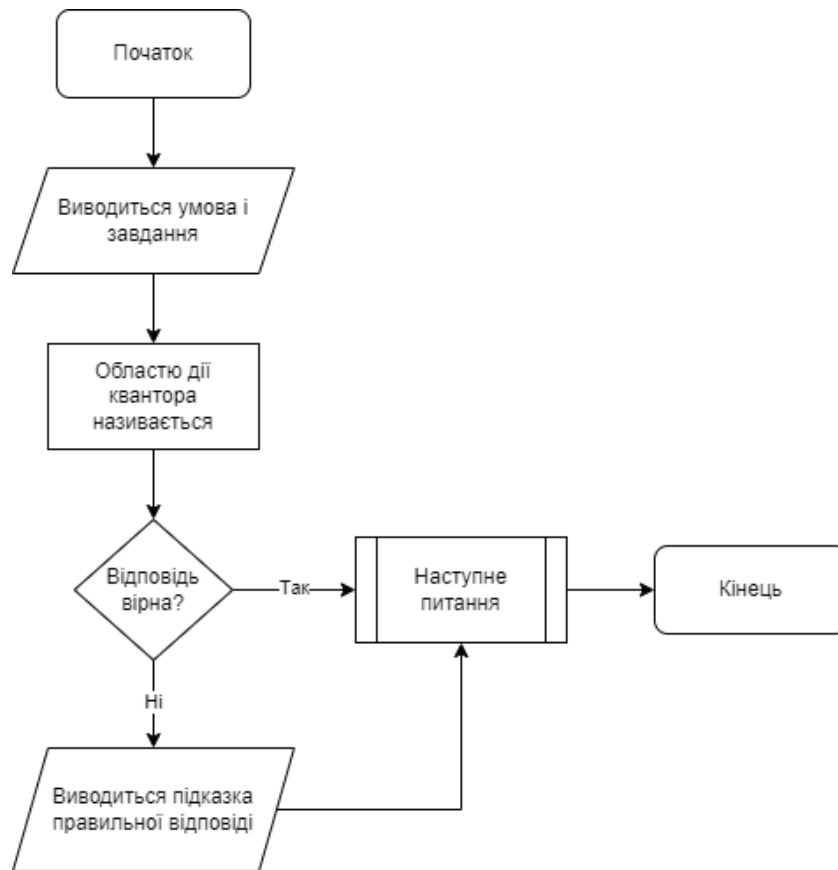


Рисунок 4.1.3 – Блок-схема другого питання тестування

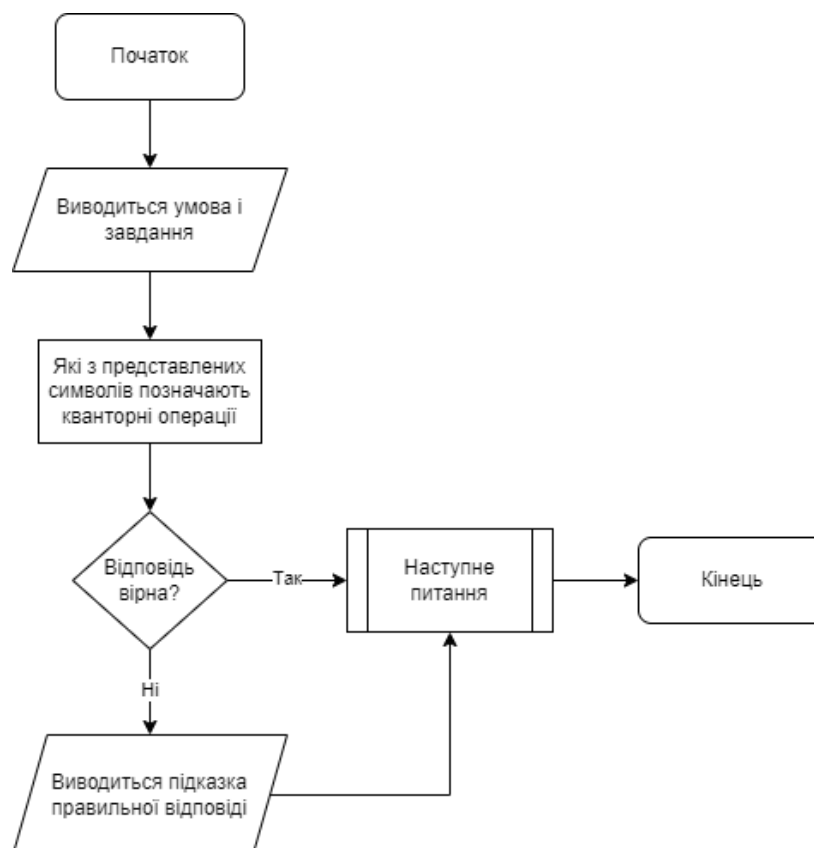


Рисунок 4.1.4 – Блок-схема третього питання тестування

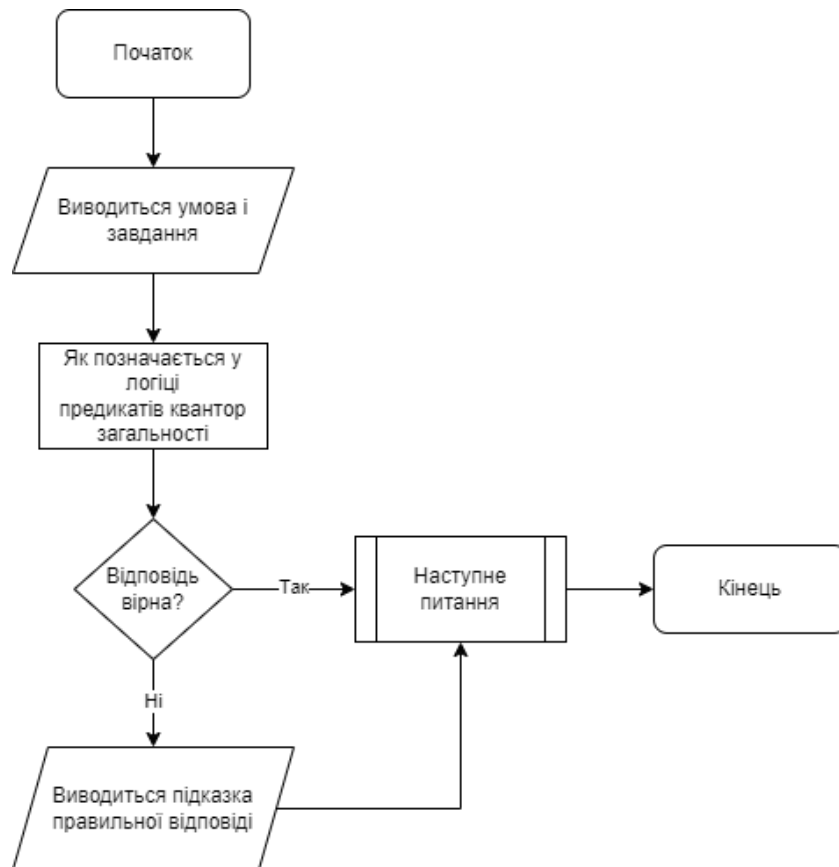


Рисунок 4.1.5 – Блок-схема четвертого питання тестування

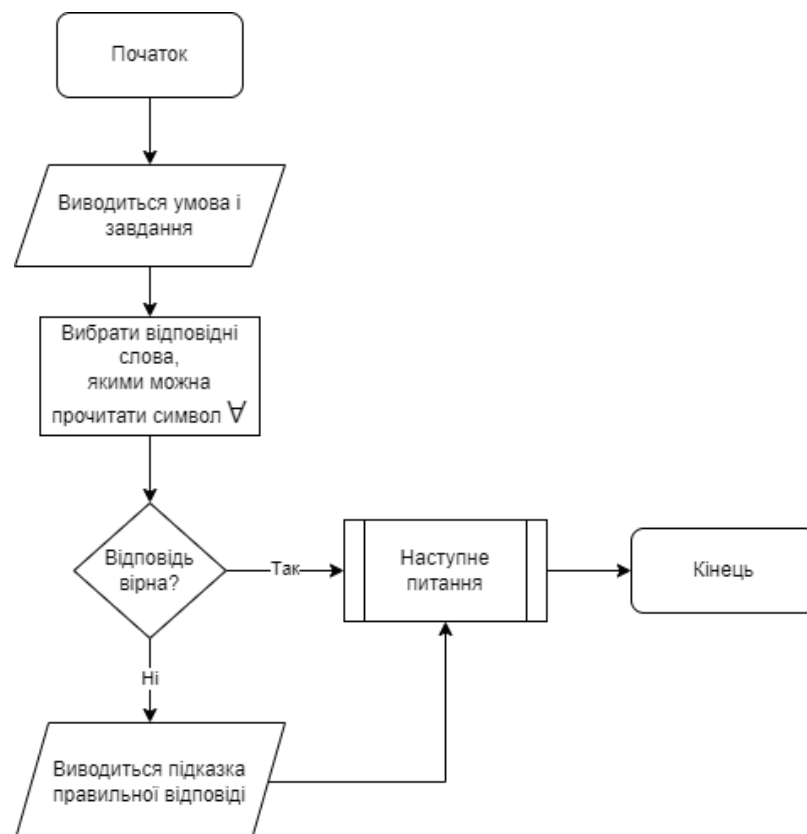


Рисунок 4.1.6 – Блок-схема п'ятого питання тестування

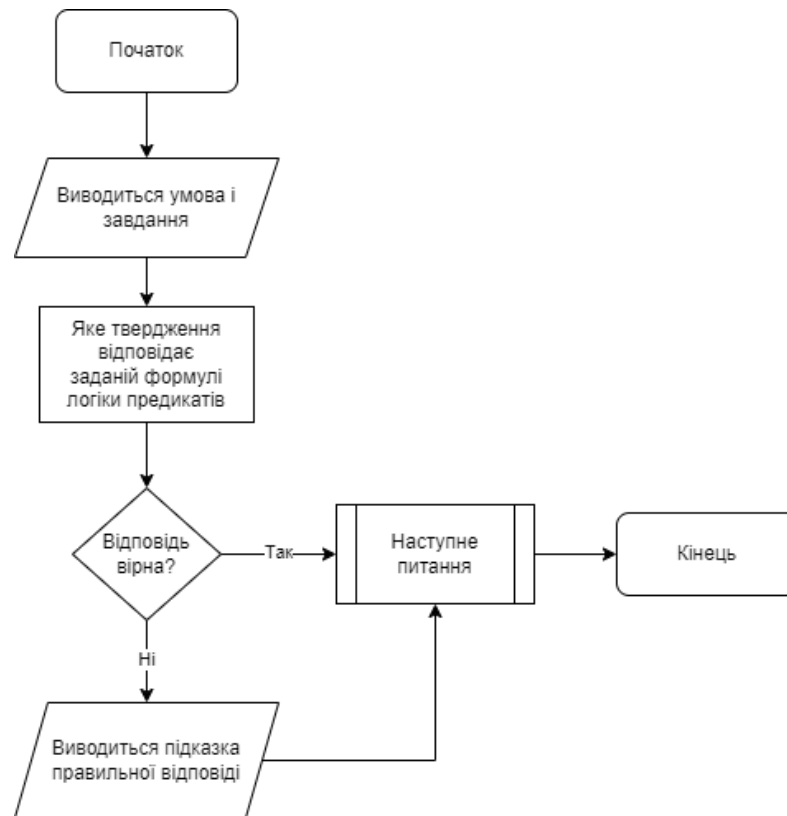


Рисунок 4.1.7 – Блок-схема шостого питання тестування

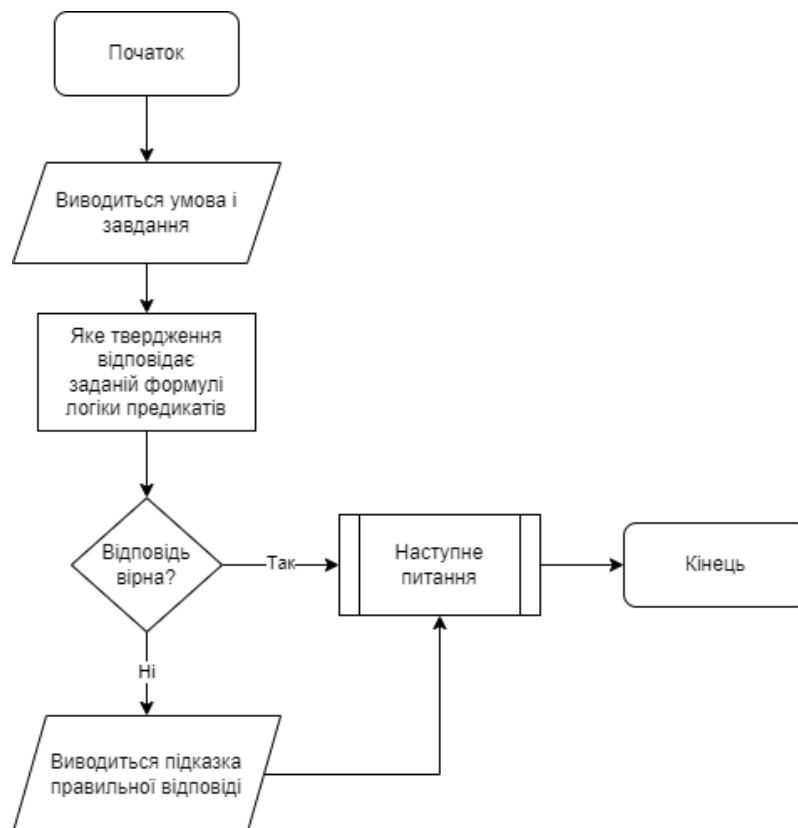


Рисунок 4.1.8 – Блок-схема сьомого питання тестування

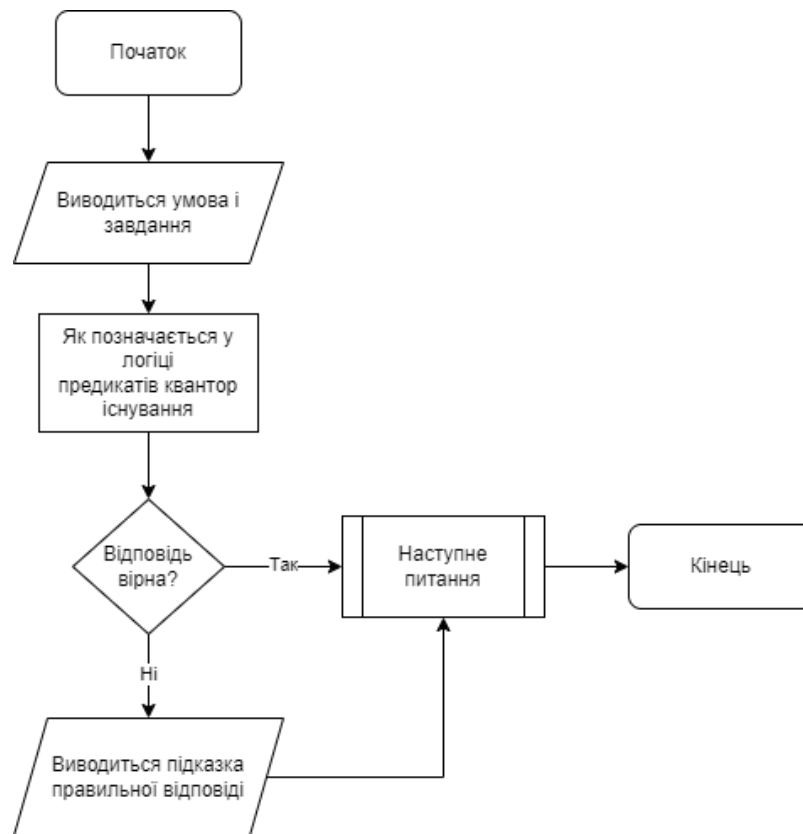


Рисунок 4.1.9 – Блок-схема восьмого питання тестування

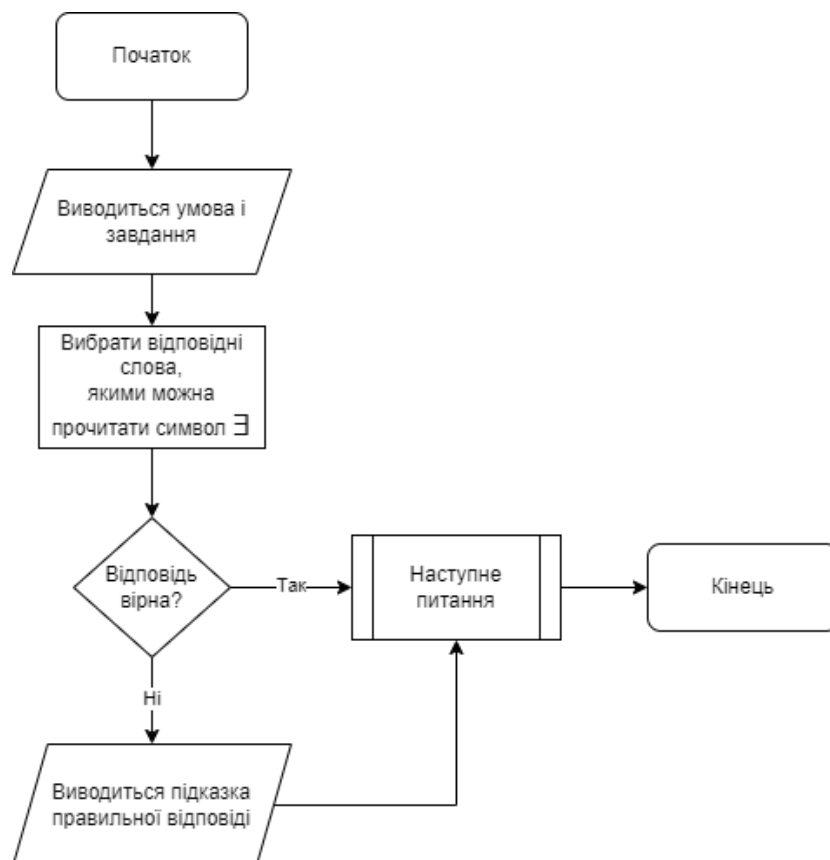


Рисунок 4.1.10 – Блок-схема дев'ятого питання тестування

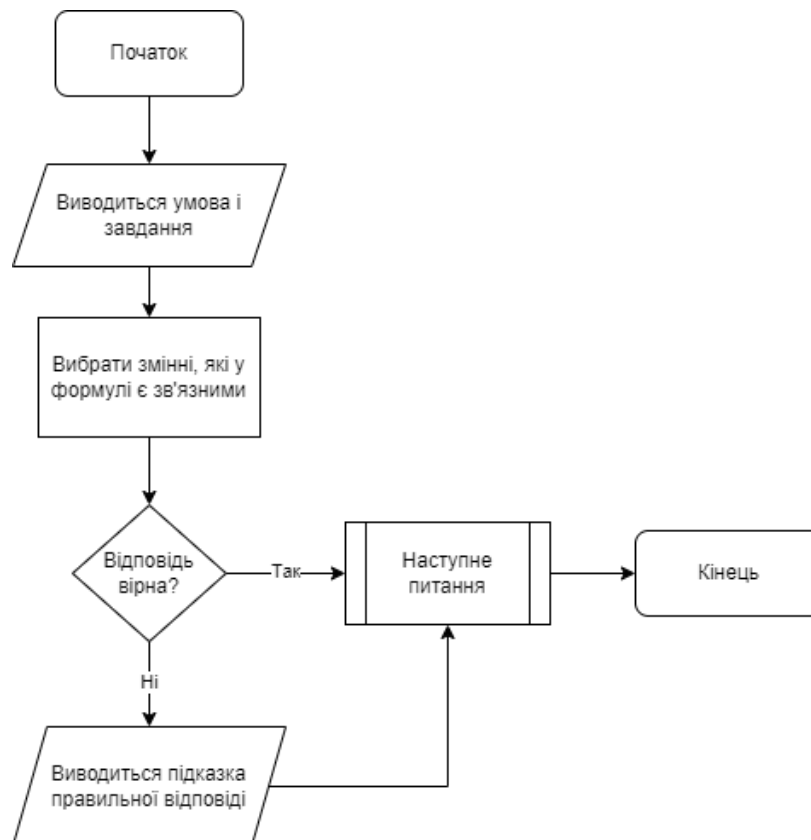


Рисунок 4.1.11 – Блок-схема десятого питання тестування

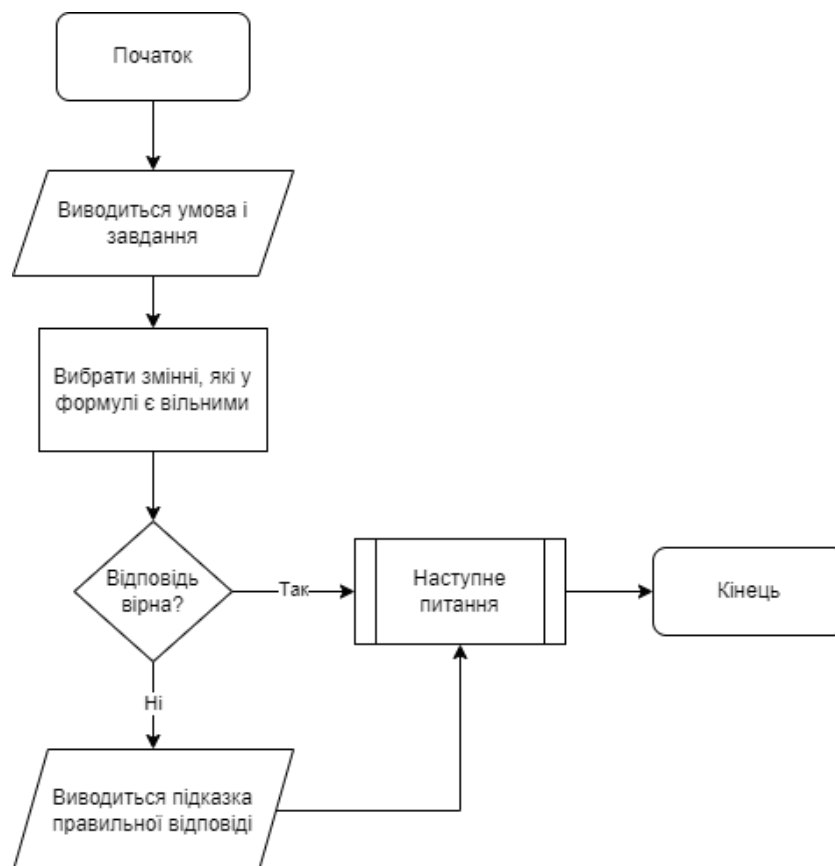


Рисунок 4.1.12 – Блок-схема одинадцятого питання тестування

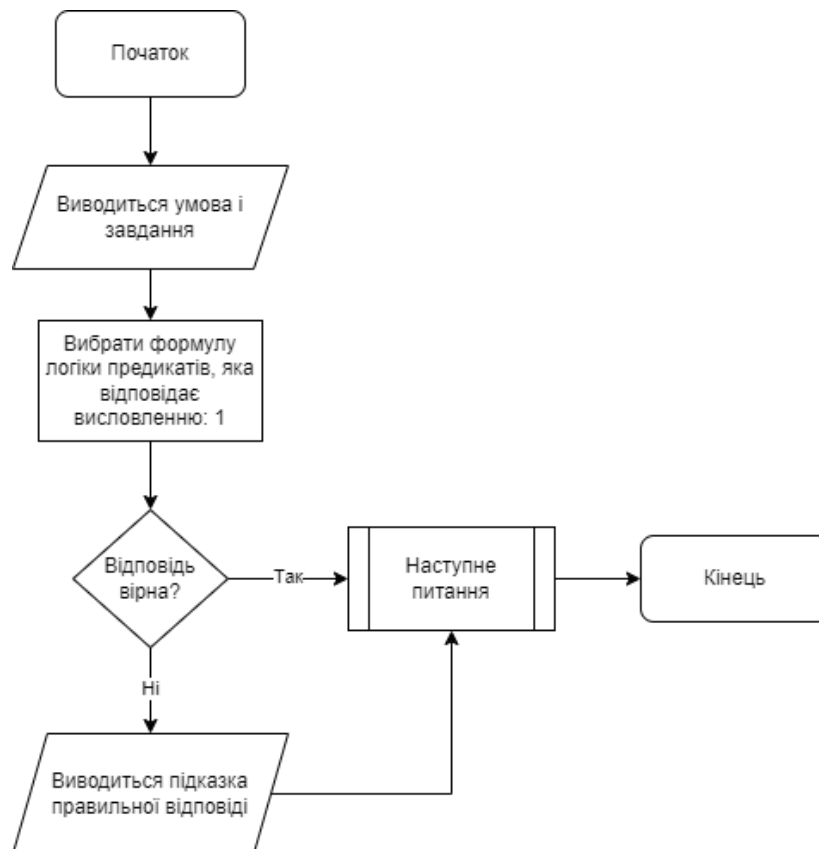


Рисунок 4.1.13 – Блок-схема дванадцятого питання тестування

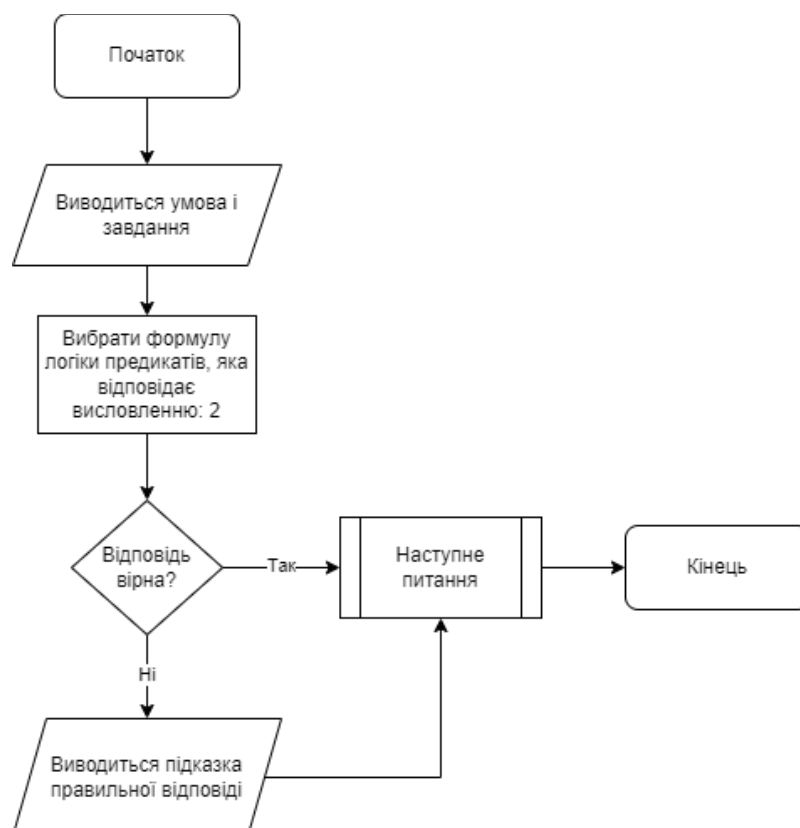


Рисунок 4.1.14 – Блок-схема тринадцятого питання тестування

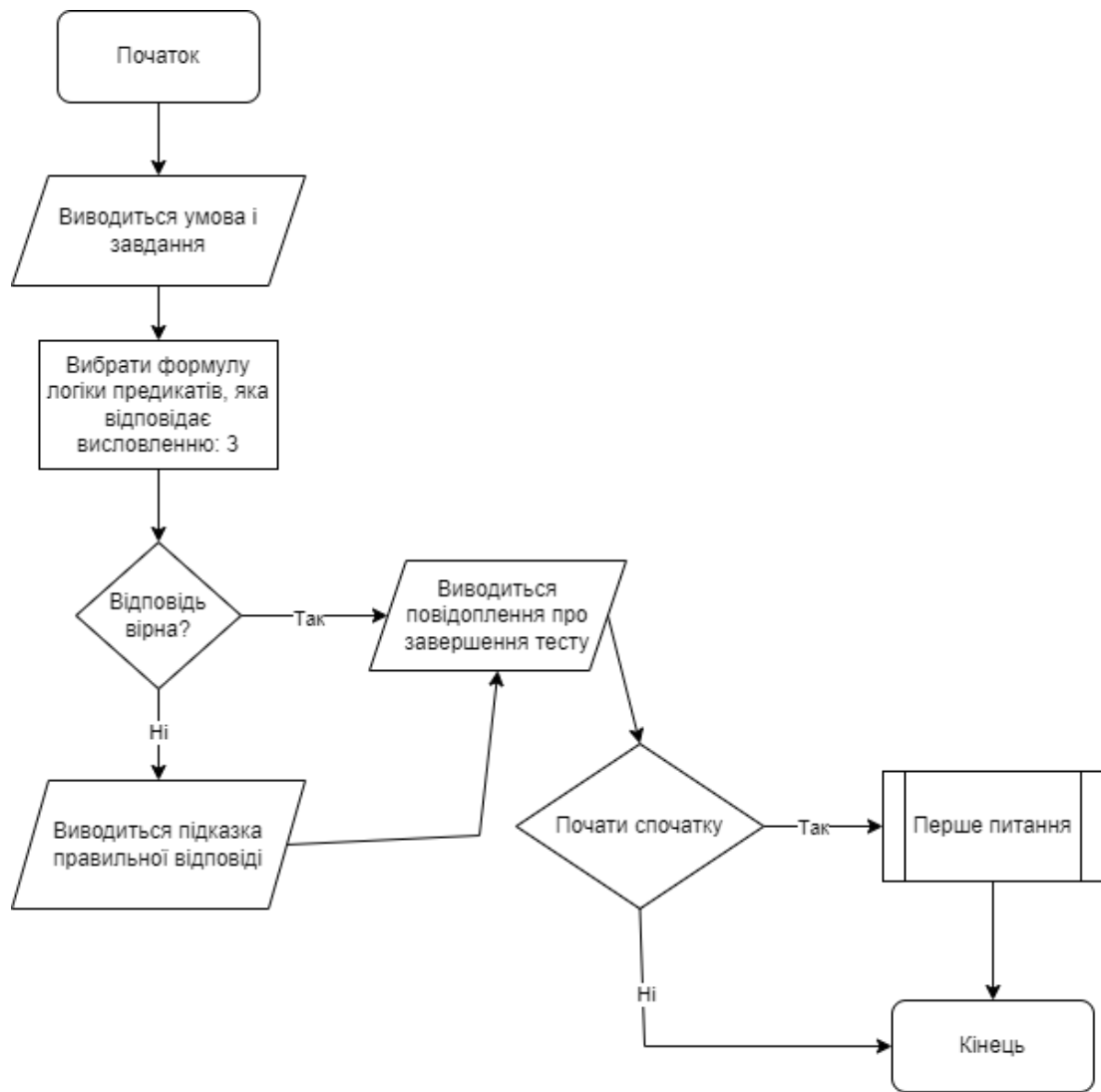


Рисунок 4.1.15 – Блок-схема чотирнадцятого питання тестування

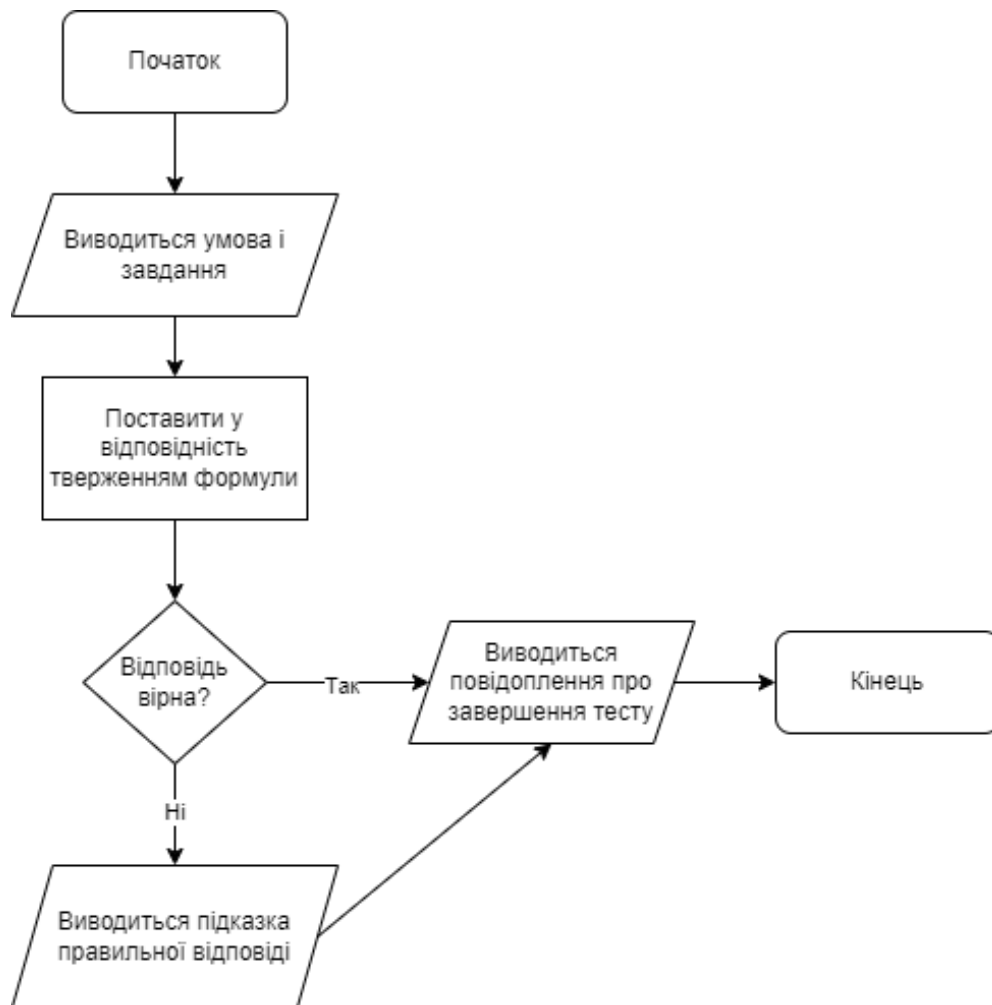


Рисунок 4.1.16 – Блок-схема практичної частини

4.2. Опис програми

Навчальний тренажер створений на мові програмування C# (.NET Framework 4.7.2) з використанням Windows Form.

«MainForm» – основна форма з якої розпочинається робота.

В даній частині коду використовується метод Start класу System.Diagnostics.Process. Цей метод відповідає за запуск нового процесу, а вказаний параметром рядок є шляхом до виконуваного файлу або ресурсу. У цьому випадку метод використовується для відкриття PDF-файлу, який містить теоретичні відомості.

```
private void TheoryButton_Click(object sender, EventArgs e)
{
    try
```

```

    {
        System.Diagnostics.Process.Start(Application.StartupPath + "\\info.pdf");
    }
    catch (Exception)
    {
        System.Diagnostics.Process.Start("https://drive.google.com/file/d/17TOpv63cjd3tDiFR-fpl5Oj0EI-
c4Loh/view?usp=sharing");
    }
}

```

«Form1» – вікно з тестуванням.

У конструкторі класу створюється список об'єктів Question за допомогою методу GenerateQuestions. Клас Question містить інформацію про питання, його варіанти відповідей та правильні відповіді.

```

private List<Question> GenerateQuestions()
{
    List<Question> questions = new List<Question>();

    questions.Add(new Question
    {
        ТекстПитання = "Квантор – це\n (1 варіант відповіді)",
        Відповіді = new List<string> {
            "логічний оператор, який описує відношення між суб'єктом та предикатом і\n несе кількісну характеристику виразу, перед яким він поставлений",
            "логічна та бітова операція, що набуває значення «істина» тоді й лише тоді,\n коли значення «істина» має тільки один з її операндів",
            "спеціальні символи для позначень вільних та зв'язаних змінних",
            "спеціальний символ, який повідомляє транслятору про те,\n що потрібно виконати операцію з деякими двома операндами"},
        ПравильніВідповіді = new List<string> { "логічний оператор, який описує відношення між суб'єктом та предикатом і\n несе кількісну характеристику виразу, перед яким він поставлений" }
    });
}

```

«Form2» – форма підсумків тестування, з можливістю почати спочатку, або перейти до головного меню.

«Form3» – вікно для практичної частини. В якій використовується ComboBox для вибору варіантів відповіді. Код перевіряючий ComboBox на унікальність:

```

private void ComboBox_SelectedIndexChanged(object sender, EventArgs e)
{
    ComboBox comboBox = (ComboBox)sender;

    // Перевірка, чи є вибраний елемент та чи він не є null
    if (comboBox.SelectedItem != null)
    {
        int обранаВідповідь;

        // Перевірка, чи вибрано число і чи воно не було обрано раніше
        if (int.TryParse(comboBox.SelectedItem.ToString(), out обранаВідповідь) &&
!обраніВідповіді.Contains(обранаВідповідь))
        {
            обраніВідповіді.Add(обранаВідповідь);
        }
        else
        {

```

```

MessageBox.Show("Будь ласка, оберіть унікальне число.");
comboBox.SelectedIndex = -1; // Знуляємо вибір, якщо вибрано не унікальне число
}
}

```

4.3. Інструкція користувачу

Запускається тренажер із ярлика програми тренажеру (рис 1):

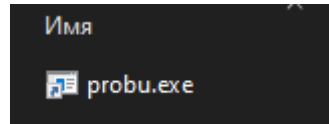


Рисунок 4.3.1 – Ярлик запуску програмного тренажеру

Після запуску тренажеру відображається головний екран програми (рис 2):

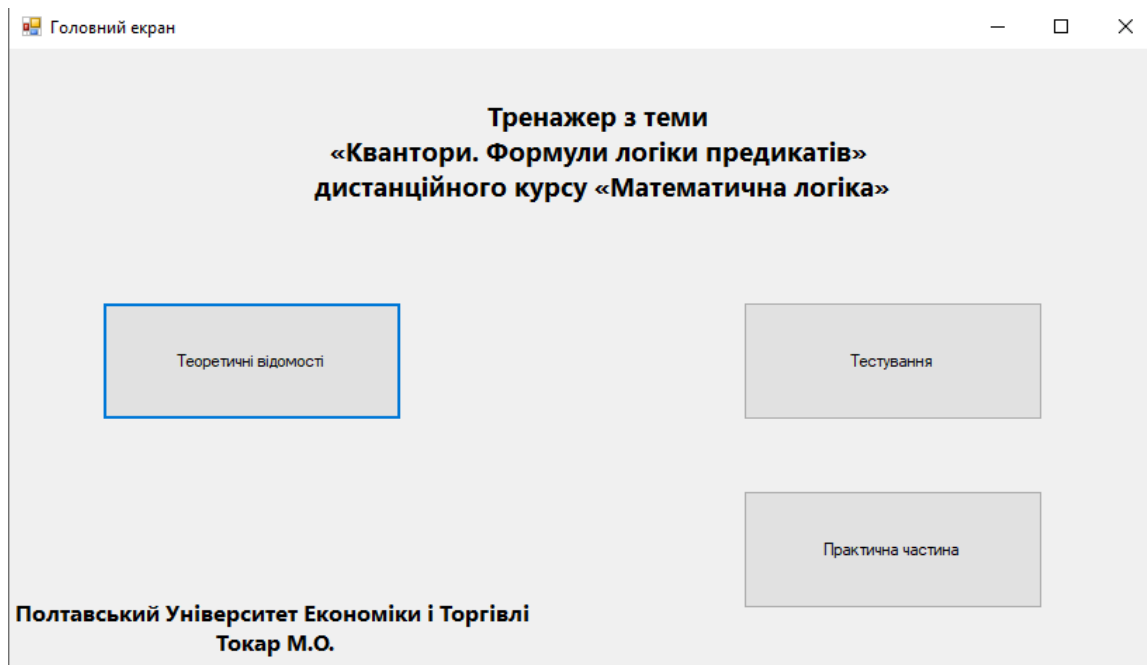


Рисунок 4.3.2 – Головний екран тренажеру

Для ознайомлення з теоретичною інформацією потрібно один натиснути на «Теоретичні відомості». Після чого відкривається PDF файл (рис 3) в якому можна ознайомитись з необхідною інформацією для проходження тестування.

3. Кванторні операції

Розглянемо операції, що перетворюють предикати у висловлення.

Нехай предикат $P(x)P(x)$ визначений на множині MM . Якщо “ aa ” – деякий елемент із множини MM , то підстановка його замість xx у предикат $P(x)P(x)$ перетворює цей предикат у висловлення $P(a)P(a)$. Таке висловлення називають одиничним. Наприклад, $R(x)R(x)$: “ xx – парне число” – предикат, а $R(6)R(6)$ – істинне висловлення, $R(3)R(3)$ – хибне висловлення.

Це ж відноситься і до nn -місних предикатів: якщо замість усіх предметних змінних $x_i, i=1, n$ підставити їх значення, то одержимо висловлення.

Поряд з утворенням із предикатів висловлень у результаті таких підстановок у логіці предикатів розглядаються ще дві операції, що перетворюють одномісний предикат у висловлення. Ці операції називаються операціями квантифікації (або просто квантифікацією, або зв'язуванням кванторами, або навішненням кванторів). При цьому розглядаються два типи так званих кванторів.

Квантор (від лат. quantum – скільки) – це логічний оператор, який описує відношення між суб'єктом та предикатом і несе кількісну характеристику виразу, перед яким він поставлений.

Квантор загальності.

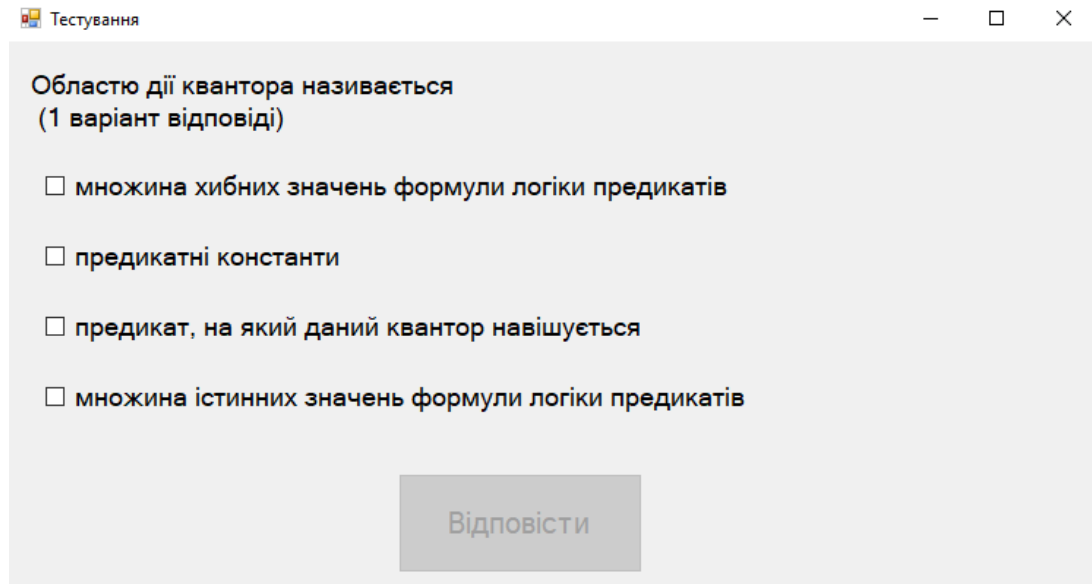
Нехай $P(x)P(x)$ – предикат, визначений на множині MM . Під виразом $\forall xP(x)\forall xP(x)$ розуміють висловлення, істинне тоді, коли $P(x)P(x)$ істинний для кожного елемента xx з множини MM , і хибний в іншому випадку. Це висловлення вже не залежить від xx . Відповідний йому словесний вираз звучить так: “Для будь-якого xx $P(x)P(x)$ істинне”. Символ $\forall$ називають квантором загальності (від нім. Alles – всі). Змінну xx у предикаті $P(x)P(x)$ називають вільною (їй можна давати різні значення з MM), у висловленні ж $\forall xP(x)\forall xP(x)$ xx називають зв'язаною квантором загальності.

Квантор існування.

Нехай $P(x)P(x)$ – предикат, визначений на множині MM . Під виразом $\exists xP(x)\exists xP(x)$ розуміють висловлення, що є істинним, якщо існує елемент $x \in M, x \in M$, для якого $P(x)P(x)$ істинне, і хибне – в іншому випадку. Це висловлення вже не залежить від xx . Відповідний йому словесний вираз звучить так: “Існує xx , при якому $P(x)P(x)$ істинне”. Символ $\exists$ називають квантором існування (від нім. Existieren – існувати). У висловленні $\exists xP(x)\exists xP(x)$ змінна xx зв'язана цим квантором. Кванторні операції застосовуються і до багатомісних предикатів. Нехай, наприклад, на

Рисунок 4.3.3 – Теоретичні матеріали

Після ознайомлення з теоретичною інформацією користувач може переходити до тесту натиснувши «Тестування». Після чого відкривається тест(рис 4) в якому обов'язково потрібно обрати хоча б один варіант відповіді, щоб перейти до наступного питання.



Тестування

Областю дії квантора називається
(1 варіант відповіді)

☐ множина хибних значень формули логіки предикатів

☐ предикатні константи

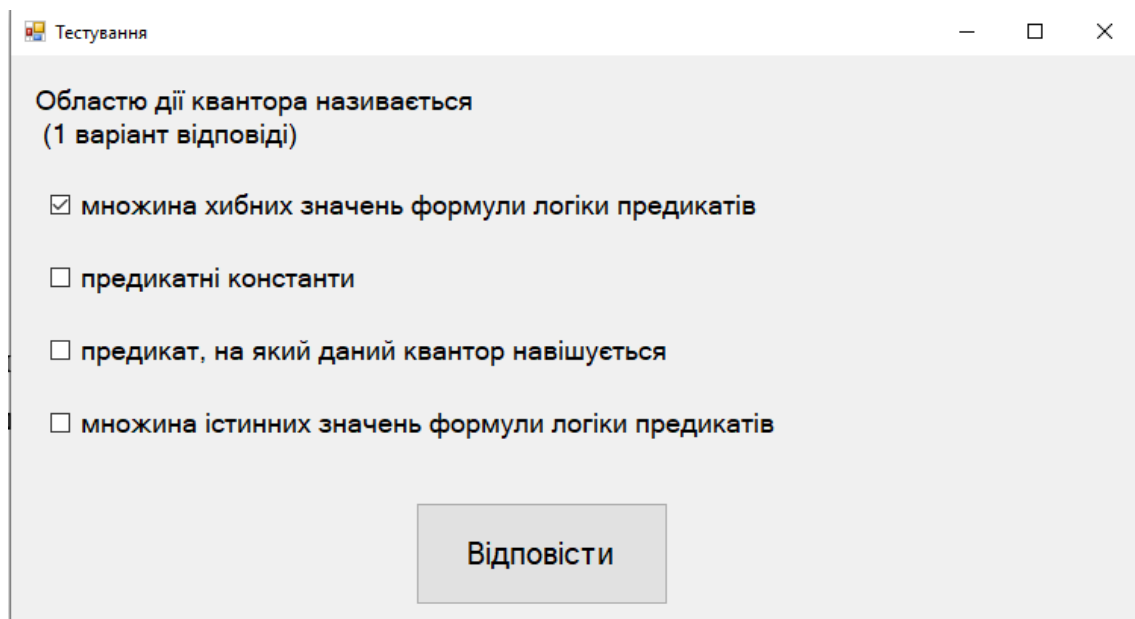
☐ предикат, на який даний квантор навішується

☐ множина істинних значень формули логіки предикатів

Відповісти

Рисунок 4.3.4 – Питання в тестуваннях

Коли було обрано хоча б один варіант відповіді вмикається кнопка «Відповісти» (рис 5). Після чого користувач може її натиснути та дізнатись чи була відповідь правильна.



Тестування

Областю дії квантора називається
(1 варіант відповіді)

☒ множина хибних значень формули логіки предикатів

☐ предикатні константи

☐ предикат, на який даний квантор навішується

☐ множина істинних значень формули логіки предикатів

Відповісти

Рисунок 4.3.5 – Кнопка відповіді

Якщо відповідь була не вірною програма повідомить про це і підкаже яка була правильною (рис 6).

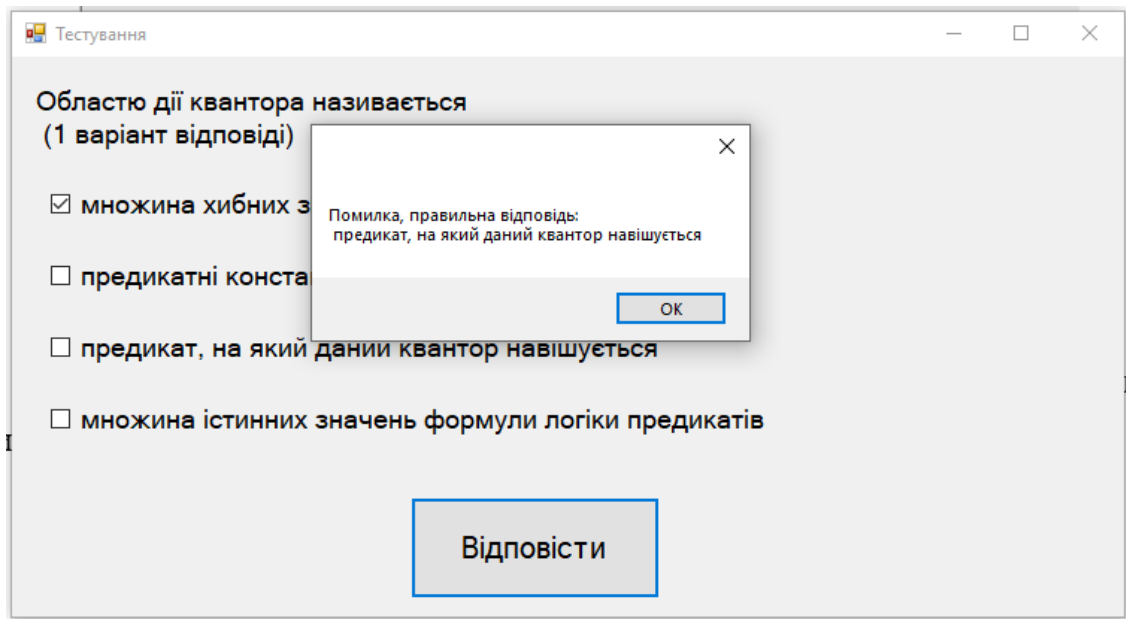


Рисунок 4.3.6 – Повідомлення помилку та підказка

Якщо відповідь була правильною програма повідомить і у користувач переходить до наступного питання(рис 7).

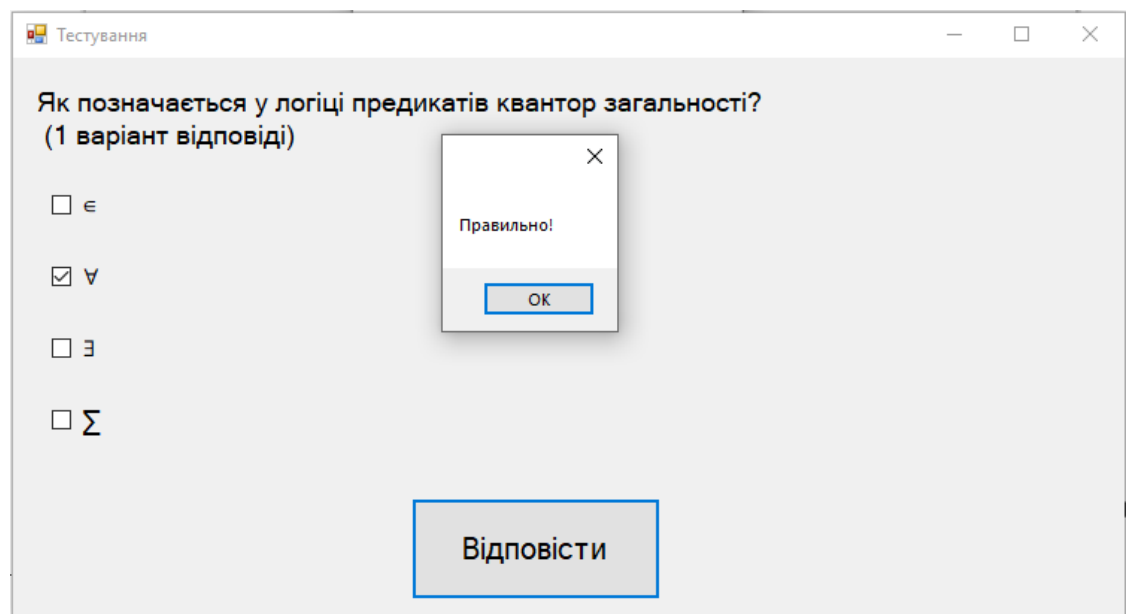


Рисунок 4.3.7 – Повідомлення правильної відповідь

Для того щоб випадково не пропустити питання в яких потрібно обрати декілька варіантів відповідей в програмі про це написана підказка (рис 8).

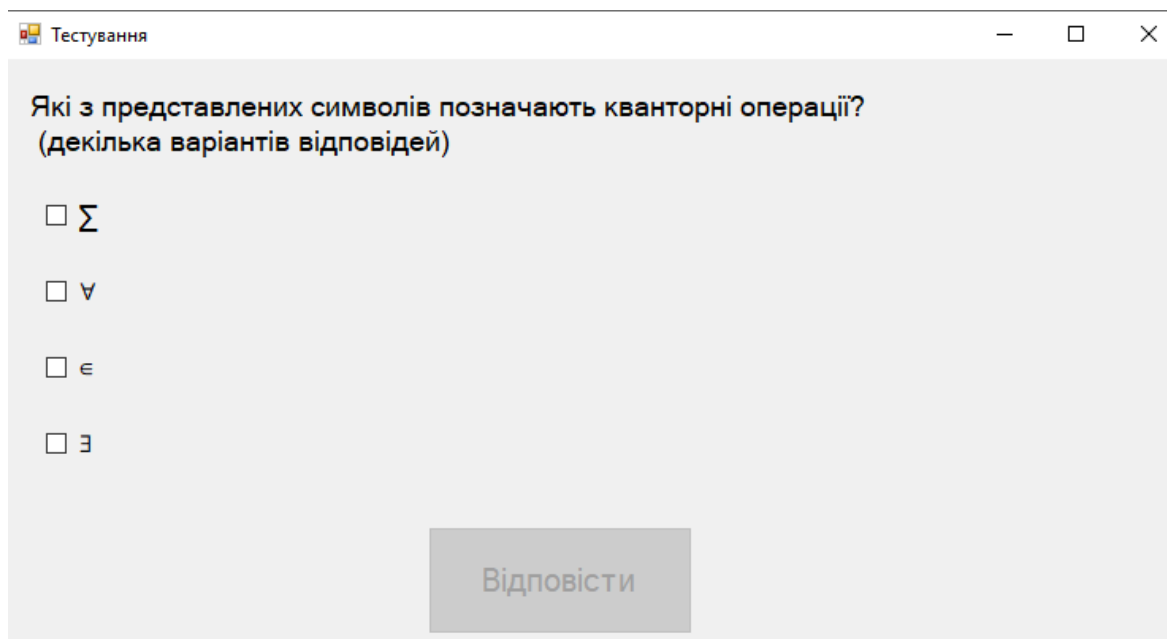


Рисунок 4.3.8 – Уточнення що в питанні декілька відповідей

Після завершення тестування користувач може побачити скільки правильних відповідей, також програма надає змогу спробувати знову або перейти до головного екрану (рис 9).

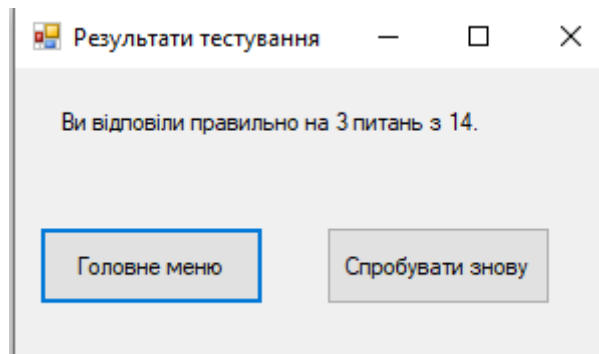
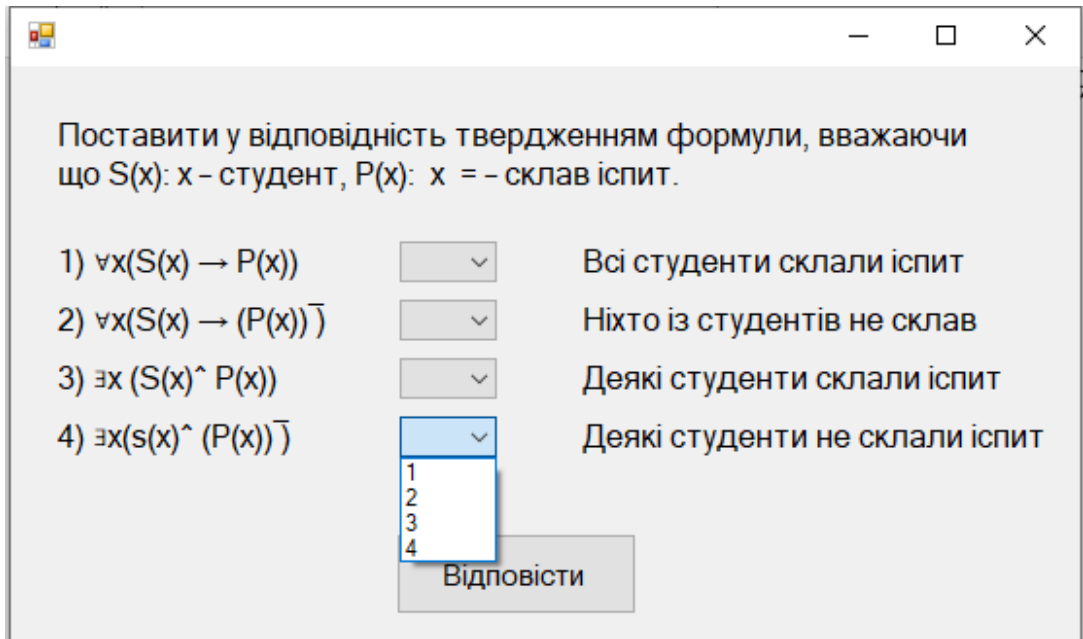


Рисунок 4.3.9 – Результати тестування

Натиснувши на «Головне меню» користувач переходить до головного екрану де може обрати «Практичну частину». Після чого відкривається завдання практичної частини (рис 10) де користувачеві потрібно встановити відповідність твердження до формул. В комірках обрати від 1 до 4.



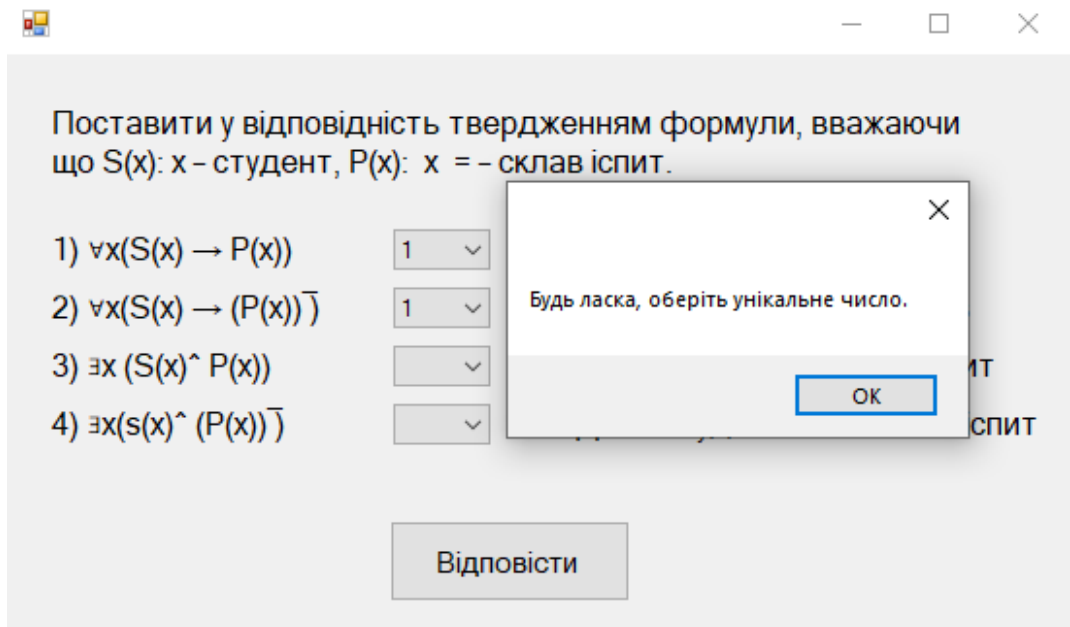
Поставити у відповідність твердженням формули, вважаючи що $S(x)$: x - студент, $P(x)$: x - склав іспит.

1) $\forall x(S(x) \rightarrow P(x))$	v	Всі студенти склали іспит
2) $\forall x(S(x) \rightarrow (P(x))\bar{>$	v	Ніхто із студентів не склав
3) $\exists x (S(x) \wedge P(x))$	v	Деякі студенти склали іспит
4) $\exists x(s(x) \wedge (P(x))\bar{>$	v	Деякі студенти не склали іспит

Відповісти

Рисунок 4.3.10 – Завдання практичної частини

Якщо користувач спробує обрати один варіант двічі програма повідомить про це (рис 11).



Поставити у відповідність твердженням формули, вважаючи що $S(x)$: x - студент, $P(x)$: x - склав іспит.

1) $\forall x(S(x) \rightarrow P(x))$	1 v	
2) $\forall x(S(x) \rightarrow (P(x))\bar{>$	1 v	
3) $\exists x (S(x) \wedge P(x))$	v	
4) $\exists x(s(x) \wedge (P(x))\bar{>$	v	

Відповісти

Будь ласка, оберіть унікальне число.

ОК

Рисунок 4.3.11 – Повідомлення потрібно обрати інший варіант

Після того як натиснули «Відповісти» програма надає статистику скільки правильних відповідей та завершується практична частина (рис 12).

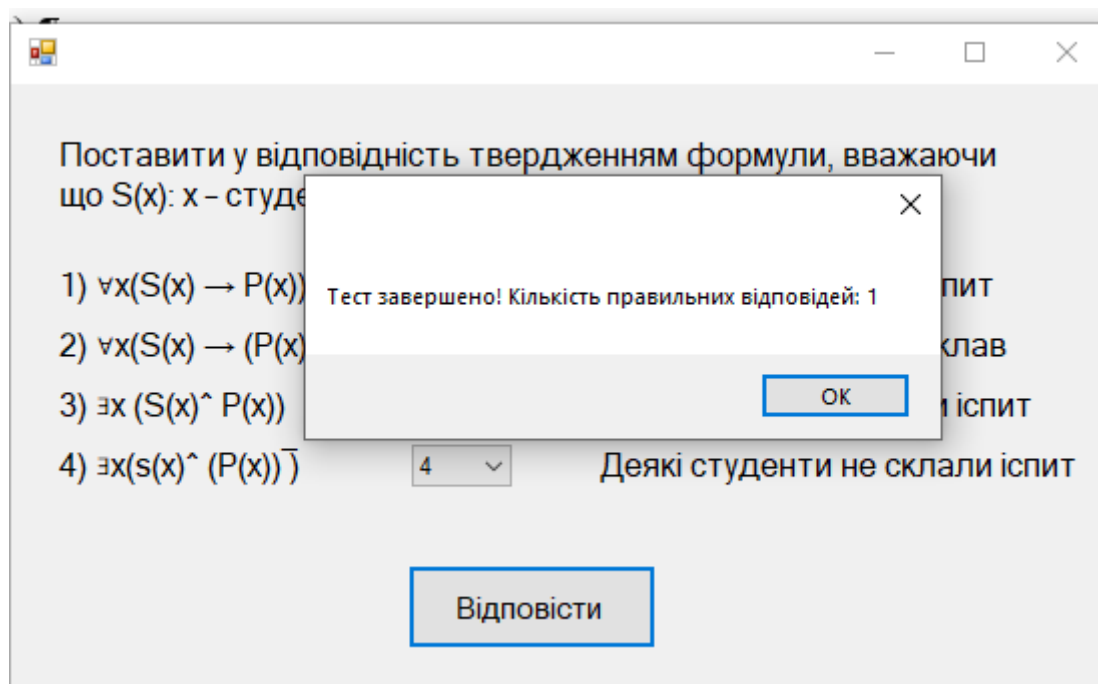


Рисунок 4.3.12 – Завершення практичної частини

ВИСНОВКИ

В результаті роботи було описано і створено навчальний тренувальний інструмент з теми «Квантори. Формули логіки предикатів» дистанційного курсу «Математична логіка».

В роботі описано такі пункти:

- Постановка завдання;
- Переваги та недоліки дистанційного навчання;
- Основні види та призначення платформ дистанційного навчання;
- Необхідність та актуальність теми;
- Основні поняття предиката, операції та побудова формул;
- Кванторні операції;
- Обґрунтування вибору мови програмування та середовища розробки;
- Алгоритм роботи програми;
- Блок-схеми алгоритму;
- Опис програми;
- Інструкція користувача.

Хоча технологічна складова навчального процесу в університетах, включаючи сучасні інформаційні технології, швидко розвивається, використання сучасних комп'ютерних технологій у навчанні не лише сприяє покращенню самостійної роботи студентів та індивідуалізації навчання, але і суттєво перетворює способи та зміст взаємодії між викладачем і студентом. Завдяки комп'ютерним технологіям, навіть при зменшенні кількості аудиторних занять, спілкування "викладач-студент" стає більш інтенсивним та активним, як у прямому, так і у зворотному напрямку, що в свою чергу підвищує якість навчання.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. О. О. Черненко. Методичні рекомендації щодо виконання кваліфікаційної роботи студентів спеціальності 122 Комп'ютерні науки освітня програма «Комп'ютерні науки» ступеня магістра/ О. О. Черненко – Полтава : ПУЕТ, 2022. – 59с.
2. Кравець В.О. Меморандум створення інформаційної освітньої мережі "Українська дистанційна освіта" / В.О. Кравець, В.М. Кухаренко [Електронний ресурс]. –Режим доступу: <http://www.osvita.org.ua/distance/ukraine/add/00/>.
3. Steps IT Academy Плюси та мінуси дистанційного навчання [Електронний ресурс] – Режим доступу: https://cloud.itstep.org/blog_3/pros-and-cons-of-distance-learning.
4. Платформи дистанційного навчання [Електронний ресурс] – Режим доступу: <https://regional-lyceum.zt.ua/pedahoham/platformy-ta-servisy-dystantsiinoho-navchannia.html>.
5. Бабій М.С. Теорія програмування: Навчальний посібник / М.С. Бабій, О.П. Чекалов.– Суми: Вид-во СумДУ, 2009. – 181 с.
6. Стівен Коул Кліні Математична логіка / Стівен Коул Кліні 2002 . – 91 с
7. Microsoft Visual Studio [Електронний ресурс] / Матеріал з Вікіпедії — вільної енциклопедії. – Режим доступу: https://uk.wikipedia.org/wiki/.NET_Framework.
8. Середовище розробки Microsoft Visual Studio – Режим доступу: https://informatics.in.ua/programming_csharp/part_01.php.
9. .NET Framework [Електронний ресурс] / Матеріал з Вікіпедії — вільної енциклопедії. – Режим доступу: https://uk.wikipedia.org/wiki/.NET_Framework.
- 10..NET Framework та C# [Електронний ресурс]. – Режим доступу: <https://www.bestprog.net/uk/2016/12/20/базові-поняття-технології-net-framework/>.
- 11.ДСТУ ГОСТ 7.1:2006 Система стандартів з інформації, бібліотечної та видавничої справи. Бібліографічний запис. Бібліографічний опис. Загальні

вимоги та правила складання / Нац. стандарт України. Вид. офіц. – [Чинний від 2007-07- 01]. Київ : Держспоживстандарт України, 2007. 47 с.

12. Положення про кваліфікаційну роботу (проєкт) (ДПСЯ М9-8.5.1-19-05-23), Полтавський університет економіки та тор-гівлі. 2023. 30 с.

ДОДАТОК А. КОД ПРОГРАМИ