

**ВИЩИЙ НАВЧАЛЬНИЙ ЗАКЛАД УКООПСПІЛКИ
«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 _____ О.ОЛЬХОВСЬКА
(підпис)
«_____» _____ 2021 р.

**ПОЯСНЮВАЛЬНА ЗАПИСКА
ДО ДИПЛОМНОЇ РОБОТИ**

**на тему
«СТВОРЕННЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТРЕНАЖЕРУ
«МЕТОД РЕЗОЛЮЦІЙ» ДИСТАНЦІЙНОГО КУРСУ
«МАТЕМАТИЧНА ЛОГІКА»**

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

Виконавець роботи Алексов Сергій Вікторович _____ «__» _____ 20__ р.
(підпис)

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

_____ «__» _____ 20__ р.
(підпис)

ПОЛТАВА 2021 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА	1
ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ	6
ВСТУП	7
1. ПОСТАНОВКА ЗАДАЧІ	9
2. ІНФОРМАЦІЙНИЙ ОГЛЯД	11
2.1. Огляд навчальних тренажерів подібної тематики	11
2.2. Позитивні та негативні аспекти огляду навчальних тренажерів	14
3. ТЕОРЕТИЧНА ЧАСТИНА	16
3.1. Метод резолюцій	16
3.2. Приклад граматики реалізації методу резолюцій, доведення в логіці висловлень	22
3.3. Уніфікація. Приклад граматики реалізації та доведення в логіці предикатів	25
3.4. Алгоритм роботи тренажера	26
3.4.1. Алгоритм застосування методу резолюцій до логіки висловлень.	27
3.4.2. Алгоритм застосування методу резолюцій до логіки предикатів.	33
3.5. Блок-схема алгоритму роботи тренажера	41
4. ПРАКТИЧНА ЧАСТИНА	46
4.1. Обґрунтування вибору програмних засобів	46
4.2. Опис процесу програмної реалізації	47
4.3. Інструкція по використанню програми-тренажеру	49
ВИСНОВКИ	54
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	55
Додаток А. Програмний код основної форми	57
Додаток Б. Вікно вибору теми	59
Додаток В. Виведення невірної довідки	61
Додаток Г. Вікно з кінцевим результатом та варіантами подальшого вибору	62

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

Умовні позначення, символи, скорочення, терміни	Пояснення умовного позначення, скорочень, символів
КНФ	Кон'юнктивна нормальна форма
тренажер	комп'ютерна програма, призначена для вивчення і закріплення різноманітних практичних навичок
C#	об'єктно-орієнтована мова програмування з безпечною системою типізації для платформи .NET.
Microsoft .NET	програмна технологія, запропонована фірмою Microsoft як платформа для створення як звичайних програм, так і веб-застосунків
Microsoft Visual Studio	серія продуктів фірми Майкрософт, які включають інтегроване середовище розробки програмного забезпечення та низку інших інструментальних засобів
.NET	крос-платформова технологія
Λ	Порожній диз'юнкт

ВСТУП

Інформаційні системи та технології заповнили усі сфери життя людини. Не оминула дану експансію і сфери освіти [5]. Нові технології вимагають нових та різноманітних методів навчання. Серед таких методів можна вважати навчально-освітні тренажери [4].

Наразі людство проходить значний етап розвитку інформаційних технологій. Не за горами впровадження сучасних кейсів та навчальних методів у середовище штучного інтелекту. Ці навчальні методики будуть сформовані для широкого спектру задач та дадуть змогу прискорити та оптимізувати машини логічного виведення. Сучасні інформаційні ресурси дозволяють вирішувати різносторонні задачі, у тому числі задачі навчання як людських індивідів так і у перспективі штучні інтелектуальні системи. Саме тому подібного роду тренажери, та як приклад тематика «Метод резолюцій» постає додатковий навчально інформаційний важіль, при навчанні. Що стосується теми «Метод резолюцій», то її актуальність полягає у розпаралелюванні вирішення задач та розробка спеціальних алгоритмів для вирішення широкого спектру завдань математичного спрямування.

Метою роботи є створення алгоритму та його програмної реалізації тренажера з теми «Метод резолюцій» дистанційного навчального курсу «Математична логіка» та закріплення знань з математичної логіки.

Об'єктом розробки в даній роботі є процес дистанційного навчання математичним дисциплінам.

Предметом розробки є програмний продукт, що реалізує тренажер з теми «Метод резолюцій» на мові програмування C# середовище Visual Studio.

Головне завдання – розробити алгоритм роботи тренажера та створити його програмну реалізацію, що відображає роботу методу резолюцій.

Перелік використаних методів полягає у застосуванні методу резолюції та програмних засобів мови C#.

Робота містить: вступ, інформаційний огляд, теоретичну частину, практичну частину, висновки та список використаних джерел.

У пункті інформаційний огляд розглянуто програми тренажери подібної тематики та математичних дисциплін.

У практичній частині було описано основні поняття, теореми та приклади по темі метод резолюцій.

У теоретичній частині розглянуто основні означення та поняття з теми «Метод резолюції» та її застосування, описано алгоритм роботи тренажера, представлено блок-схему алгоритму роботи програми.

Особистий внесок було здійснено у практичній частині – обґрунтування вибору мови програмування, створення алгоритмів задач, створення програмного коду та інструкція з використання тренажера. Написання тренажера згідно раніше розробленого алгоритму задач.

1. ПОСТАНОВКА ЗАДАЧІ

Виходячи з того, що є необхідність розробити навчальний тренажер з теми «Метод резолюцій» до дистанційного курсу «Математична логіка», слід виконати наступні поставлені задачі:

- Розглянути та провести аналіз теоретичних відомостей по методу резолюцій;
- Розглянути особливості використання методу резолюцій у логіці висловлень;
- Розробити алгоритм тренажеру по темі: «Метод резолюцій»;
- Обрати та обґрунтувати вибір програмного забезпечення, для створення тренажеру;
- Розробити блок схему роботи програмного середовища тренажеру;
- Описати процес програмної реалізації;
- Описати середовище та принцип роботи тренажеру.

У навчальному тренажері буде реалізована сторінка із теоретичною інформацією, та тестування з теоретично-практичного змісту. Також буде тестування, покрокове рішення задач методом резолюцій, що буде застосований до логіки висловлень та логіки предикатів. При проходженні тестування, користувач завжди матиме доступ до теоретичного матеріалу, що допоможе йому аналізувати свої помилки одразу.

Побудувати алгоритми рішення задач методом резолюцій, та створити блок-схеми тренажеру. Та на основі алгоритму створити програмний тренажер який реалізовував би рішення задач математичного спрямування, а саме «Методу резолюцій».

Тренажер міститиме:

1. Стартова сторінка.
2. Сторінка із теоретичними матеріалами.

3. Сторінку із тестуванням по методу резолюцій.
4. Сторінку з умовою практичного завдання (прикладу), покроковими завданнями та варіантами відповідей.
5. Кінцеву сторінку із результатами проходження та вибором подальших дій.

Сторінка із теоретичним матеріалом буде доступна користувачу увесь час, щоб користувач міг звернутись до неї у будь-який момент.

Якщо користувач не вірно відповість на поставлене запитання, перед ним з'явиться повідомлення про помилку.

При завершенні тестування, буде реалізована можливість користувачу пройти тренажер повторно, обрати інше тестування, або завершити його.

2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Огляд навчальних тренажерів подібної тематики

Тренажер з теми «Метод резолюцій» дистанційного навчального курсу «Математична логіка». На першій сторінці (стартовій) виводиться вибір мови тренажеру, тема, кнопки: теоретичної, практичної частин та вихід. В нижній частині тренажеру розміщений допис з наступною інформацією:

Назва навчального закладу, автор , керівник та тема.

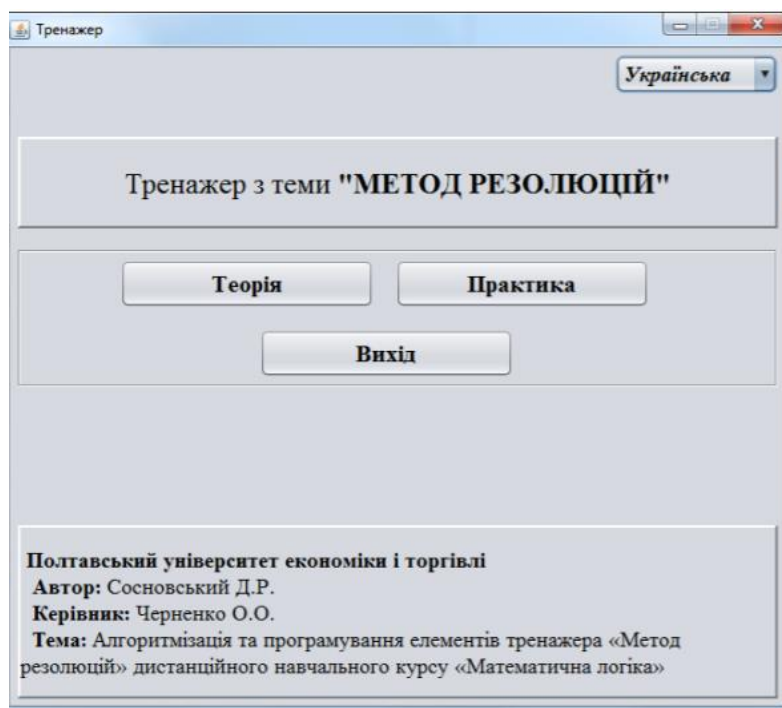


Рисунок 2.1 – Стартове середовище тренажеру

При натисканні на кнопку теорія відбувається перехід на теоретичні відомості, де систематизована теоретична інформація.



Рисунок 2.2 – Панель з теоретичним матеріалом
Кожний приклад розпочинається з виводу умови.

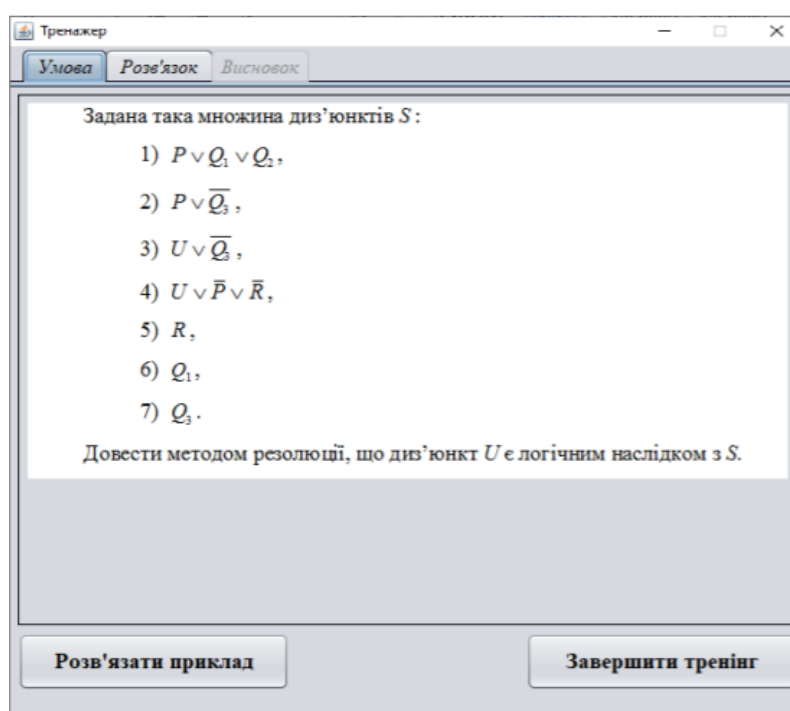


Рисунок 2.3 – Вікно умови завдання

На кожному кроці виводиться умова, завдання та варіанти відповіді. Також розташовані дві кнопки: одна – звернення до теорії, інша – перехід на наступний крок.

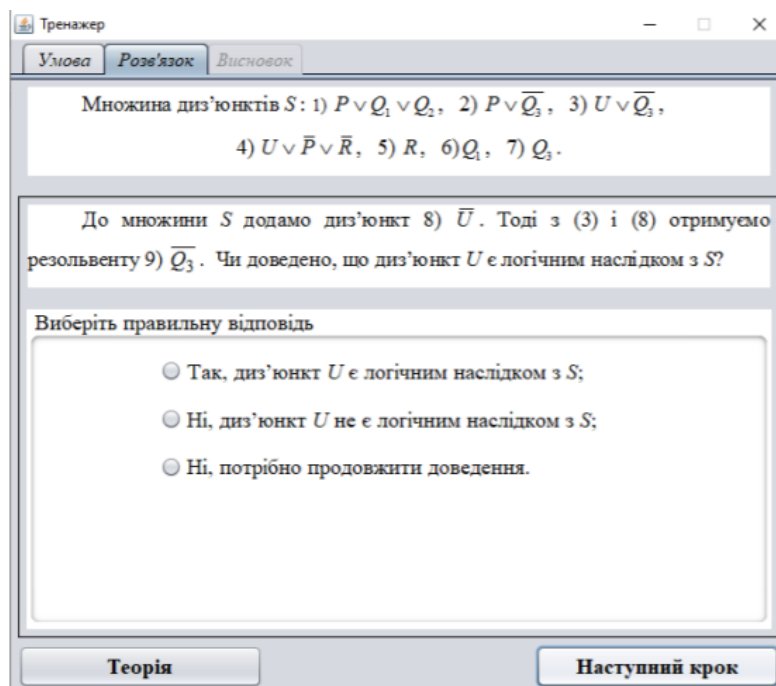


Рисунок 2.4 – Один із кроків тренажеру

При неправильній відповіді на будь-якому кроці відображається повідомлення про помилку.

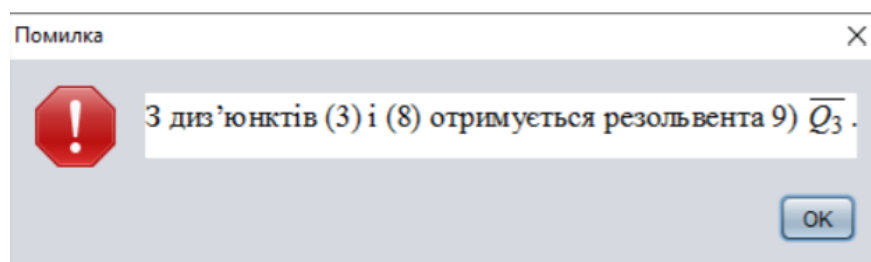


Рисунок 2.5 – Повідомлення про невірну відповідь та підказка

Після завершення тренажеру з'являється вікно із висновками проходження.

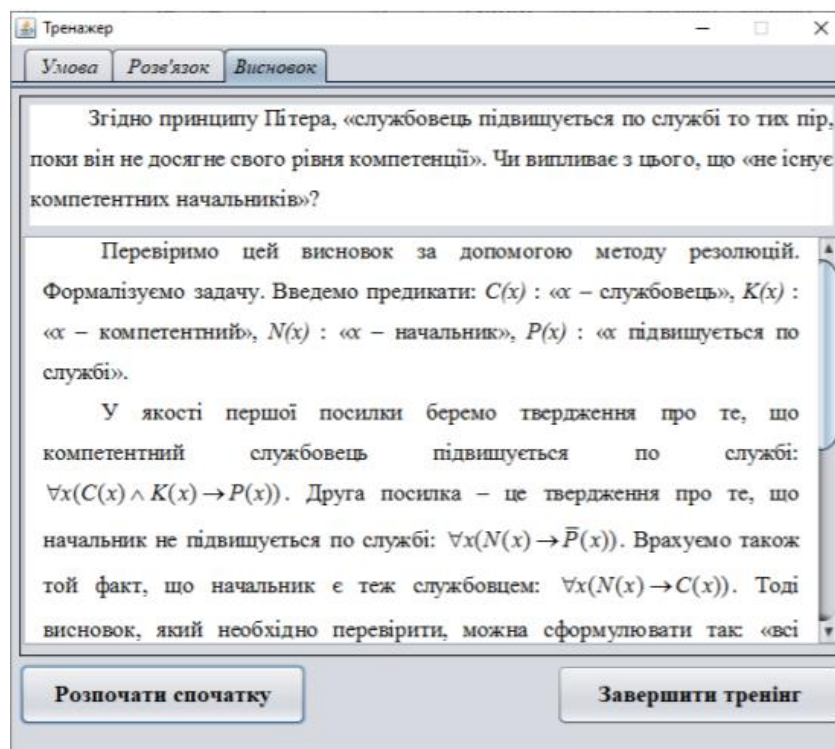


Рисунок 2.6 – Висновок тренажеру

Також було розглянуто, ще декілька тренажерів математичних дисциплін. Серед розглянутих тренажерів, були тренажери наступної тематики: «Сортування бульбашки» дистанційного навчального курсу «Аналіз алгоритмів», «Системи числення» дистанційного навчального курсу «Інформаційні мережі», «Графічні методи розв'язування задач лінійного програмування» дистанційного навчального курсу «Методи оптимізації та дослідження операцій».

2.2. Позитивні та негативні аспекти огляду навчальних тренажерів

Розглянуті тренажери мають свої як плюси та і мінуси.

До плюсів можна віднести:

- наявність умови на кожному кроці тренажеру;
- покроковий розв'язок;
- вся структура систематизована;
- можливість звернутись до теорії на будь-якому кроці;
- можливість повторного проходження;

- інформацію про помилку при невірній відповіді.

До мінусів можна віднести:

- відсутність опису тренажеру;
- недостатньо інформації у повідомленні про помилку
- у висновку відсутня інформація про кількість набраних балів.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Метод резолюцій

Метод резолюцій спирається на обчислення резольвент. Існує теорема, яка стверджує, що питання про доказовість довільної формули в численні предикатів зводиться до питання про вивідність порожнього списку в обчисленні резольвент. Тому доведення того, що список формул в обчисленні резольвент порожній, еквівалентне доведенню хибності формули в численні предикатів.

Доказ подібних теорем зводиться до доказу того, що деяка формула G (власне гіпотеза теореми) є логічним множин формул $F_1, \dots, F_k$ (припущення). Тобто сама теорема має можливість бути сформульованою наступним чином: «якщо $F_1, \dots, F_k$ істинні, то істинною є й G .

Для доказу того, що формула G є логічним наслідком множин формул $F_1, \dots, F_k$, метод резолюції застосовується наступним чином. Спочатку складається множина формул $F_1, \dots, F_k, \overline{G_g}$. Потім кожна з формул приводиться до КНФ (сполучення диз'юнктивів) і в отриманих формулах закріплюються знаки кон'юнкції. Виходить безліч диз'юнктивів S . І, нарешті, шукається висновок порожнього диз'юнктивів з S . Якщо порожній диз'юнктив виводимо з S , то формула G є логічним наслідком формул $F_1, \dots, F_k$. Ось такий метод доведення теорем, називається методом резолюцій.

Кон'юнктивна нормальна форма (КНФ) в булевій логіці – нормальна форма в якій булева формула має вид кон'юнкції декількох диз'юнктивів (де диз'юнктами називаються диз'юнкції декількох пропозиційних символів або їх заперечень). Кон'юнктивна нормальна форма широко використовується в автоматичному доведенні теорем, зокрема вона є основою для використання правила резолюції.

Приклад формули записаної в КНФ:

$$A_1 \vee A_2;$$

$$A_1;$$

$$(A_1 \vee A_2) \wedge \overline{A_1};$$

$$(A_1 \vee A_2 \vee \overline{A_3}) \wedge (\overline{B_1} \vee B_2 \vee B_3) \wedge (A_3 \vee B_1) \wedge A_2;$$

Наступні формули не є КНФ:

$$\overline{(A_2 \vee A_3)};$$

$$(A_1 \vee A_2) \wedge A_3;$$

$$A_1 \wedge (A_2 \vee (B_1 \vee B_2));$$

Проте ці формули еквівалентні наступним формулам записаним у кон'юнктивній нормальній формі:

$$\overline{A_2} \wedge A_3;$$

$$(A_1 \vee A_3) \wedge (A_2 \vee A_3);$$

$$A_1 \wedge (A_2 \vee B_1) \wedge (A_2 \vee B_2);$$

Правило резолюцій — це правило висловлювання, що сходиться до методу доказу теорем через пошук протиріч; використовується в логіці висловлювань і логіці предикатів першого порядку. Правило резолюцій, що застосовується послідовно для списку резольвент, дозволяє відповісти на питання, чи існує у вхідній множині логічних виразів протиріччя. Правило резолюцій запропоновано в 1930 році в докторській дисертації Жака Ербрана для доведення теорем у формальних системах першого порядку. Правило розроблено Джоном Аланом Робінсоном в 1965 році.

Логіка першого порядку (числення предикатів) — це формальна система в математичній логіці, в якій допускаються висловлення відносно змінних, фіксованих функцій, і предикатів. Є розширенням логіки висловлювань. В свою чергу є частковим випадком логіки вищого порядку[2].

Логіка висловлень — формальна система в математичній логіці, в якій формули, що відповідають висловленням, можуть утворюватись шляхом з'єднання простих висловлень із допомогою логічних операцій, та система правил виводу, які дозволяють визначати певні формули як «теореми» формальної системи.

Означення. Елементарну формулу логіки висловлень, або її заперечення, називають *літерою*. Літери будемо позначати прописними латинськими літерами.

Приклад.

Формули $a=P$, $b=\overline{P}$ є літерами, а формула $P \rightarrow Q$ не є літерою.

Диз'юнкція — двомісна логічна операція, що має значення «істина», якщо хоча б один з операндів має значення «істина». Операція відображає вживання сполучника «або» в логічних висловлюваннях. Диз'юнкція є бінарною операцією, тобто має два операнда. Запис може бути префіксним (знак операції стоїть перед операндами), інфіксним (знак операції стоїть між операндами) або постфіксним (знак операції стоїть після операндів). За кількості операндів понад 2 префіксний і постфіксний записи економніші[3].

Позначається: в математиці та логіці як $\wedge$, у програмуванні як $|$ чи *or*. Найчастіше трапляються такі варіанти запису: $a||b$, $a|b$, $a \vee b$, $a+b$, $a \text{ or } b$.

Означення. Скінчення диз'юнкція літер або їх заперечення називається диз'юнктом. Порожній диз'юнкт позначається $\wedge$ та означає хибність, або тотожно рівну нулю формулу.

Приклад. Диз'юнкції: $a \vee b$, $a \vee \overline{b}$, $p \vee q \vee r$, $\overline{p}$ є диз'юнктами, а формула $a \vee (\overline{p \vee q})$ – ні.

Означення. Два диз'юнкта називаються резольвентною парою, якщо існує така літера, яка бере участь в одному диз'юнкті як додатна (записана у прямому вигляді), а в другому - як від'ємна (записана у інверсному вигляді).

Приклад. Пари диз'юнктів $a \vee b$, $a \vee \overline{b} \vee c$ – *резольвентна*, так як літера b бере участь у першому диз'юнкті як додатна, а в другому – як від'ємна. Пара $a \vee b$, $a \vee c$ не є резольвентною парою.

Теорема. Нехай d_1 , d_2 – резольвентна пара диз'юнктів вигляду $d_1 = a \vee c$, $d_2 = b \vee \overline{c}$, де через a , b позначені члени диз'юнктів з невиділеними літерами.

Тоді формула $a \vee c \wedge b \vee \bar{c} \rightarrow a \vee b$ є *тофтологією*, тобто тотожно-істиною, чи логічним законом.

Доведення. Якщо посилка в імплікації – хибна, то формула – істинна. Якщо посилка істинна, то кожна думка $(a \vee c), (b \vee \bar{c})$ – істинна.[2]

Можливо два випадки: c – істинна, c – хибна.

В першому випадку буде: b – істинна і, адже, наслідок імплікації $a \vee b$ – істинна. Так як посилка та наслідок імплікації – істинна, то вся формула імплікації – істинна.

В другому випадку буде: a – істинна і, отже, знову наслідок $a \vee b$ – істинна. Так як посилка та наслідок імплікації – істинна, то вся формула імплікації – істинна.

Із цієї теореми слідує правило отримання із резольвентної пари нового диз'юнкту, який називається резольвентою. Це правило називається правилом резолюції та його записують у вигляді:

$$\frac{(a \vee c), (b \vee \bar{c})}{a \vee b}$$

Приклад. Резольвентою для пари $a \vee b \vee c, q \vee \bar{c}$ буде диз'юнкт $a \vee b \vee q$. Застосовуємо правило резолюції до пари $r \vee \bar{h} \vee t, h \vee t$. Отримаємо резольвенту $r \vee t$.

Метод резолюції відноситься до напівконструктивного методу, він легко піддається алгоритмізації. Суть його полягає в тому, що два посилкових диз'юнкта з протилежними термами завжди можна склеїти в один заключний диз'юнкт, в якому відсутні протилежні терми

$$A \vee B, C \vee \bar{B} = A \vee C$$

Принцип резолюції повністю змінює аксіому порядку, оскільки вона сама може бути доведена в рамках методу резолюції.

$$A, B \rightarrow A;$$

$$A, B, \bar{A} \rightarrow 0;$$

$$0, B \rightarrow 0;$$

Звернемо увагу на те, що посилка В взагалі не використовується, тобто необов'язково треба використовувати всі посилки, головне отримати нуль. Всі доведення клауз починають з приведення їх в нормальну кон'юнктивну форму, потім виписують по порядку всі посилки і склеюють згідно з чергою.

Проаналізуємо покроково доведення в алгебрі висловлень методом резолюції:

1. Нехай треба довести в алгебрі висловлень, що формула H – тавтологія (тобто $H = 1$).

2. Розглянемо заперечення цієї формули: $G = H$. Тоді задача переформулюється та стає наступною. Довести, що формула G – тотожно хибна (G).

3. Перетворимо формулу G в кон'юнктивну нормальну форму (КНФ)

$$G = \bigwedge d_i = d_1 \wedge d_2 \wedge \dots \wedge d_n,$$

де d_i диз'юнкти.

4. Серед диз'юнктивів d_i знайдемо резольвентну пару і застосуємо до неї правило резолюції. Отриманий новий диз'юнкт, резольвенту, позначимо d_{n+1} та додамо у формулу КНФ для G .

$$G = \bigwedge d_i = d_1 \wedge d_2 \wedge \dots \wedge d_n \wedge d_{n+1},$$

Таким чином отримаємо нову КНФ для G .

5. Якщо резольвента $d_{n+1} = \bigwedge$ є порожнім диз'юнктом, то формула G буде тотожно хибною, задача розв'язана та відбувається зупинка. Якщо ні, то знову знайде, як в пункті 4., серед всіх диз'юнктивів $d_1 d_2, \dots, d_n d_{n+1}$ резольвентну до існуючих диз'юнктивів. Повторюватиметься пункт 4 та 5, до тієї пори, поки не отримаємо порожній диз'юнкт.

Теоретичним поясненням методу резолюцій можна вважати наступні теореми:

Теорема. (резольвентна пара)

Нехай формула G – тотожно хибна та представлена та представлена КНФ $G = d_1 \wedge \dots \wedge d_n$. Тоді серед диз'юнктивів d_i існує резольвентна пара.

Теорема. (додавання резольвент)

Нехай формула представлена в КНФ $G = d_1 \wedge \dots \wedge d_n$ та серед диз'юнктивів d_i існує резольвентна пара. Тоді додавання у формулу КНФ резольвенти d_{n+1} даної пари є рівносильним перетворенням формули G , тобто:

$$G = d_1 \wedge \dots \wedge d_n \rightarrow G \equiv d_1 \wedge \dots \wedge d_n \wedge d_{n+1}$$

Теорема 4. (порожня резольвента)

Якщо формула – тотожно-хибна та представлена в КНФ, то серед всіх резольвент вихідних диз'юнктивів та знову отриманих диз'юнктивів за правилом резолюції існує порожній диз'юнкт.

Подальшим спрощенням методу резолюцій є ідея американського математика Хорна використовувати тільки диз'юнкти певного вигляду.

Хорнівським диз'юнктом називається диз'юнкт не більше ніж з одним позитивним літералом[2].

Означення. Диз'юнкт у якого серед літер не більше однієї додатної називається хорнівським. При цьому диз'юнкт з однією додатною літерою і наявністю від'ємних літер виглядає:

$$a \vee \bar{b} \vee \dots \vee \bar{c} \equiv (b \wedge \dots \wedge c) \rightarrow a$$

називається правилом, та може бути прочитане так: «із $b, \dots, c$ слідує a ».

Диз'юнкт без додатної літери вигляду:

$$\bar{b} \vee \dots \vee \bar{c} \equiv (b \wedge \dots \wedge c) \rightarrow \Lambda$$

називається цільовим диз'юнктом «із $b, \dots, c$ слідує хибність»

Диз'юнкт без від'ємних літер тільки з однією додатною літерою виглядає:

$$a \equiv (I \rightarrow a)$$

називається фактом та читається «справедливо a ».

Висновок. Для того щоб довести методом резолюцій, що формула G тотожно-хибна, рекомендовано виконати ряд дій.

1. Привести G до КНФ.
2. Перетворити усі диз'юнкти у КНФ в хорнівським шляхом перепозначенням деяких додатних літер у від'ємні.

Якщо всі диз'юнкти в КНФ – хорнівські, то застосування методу резолюцій полегшується тим, що завжди можна звести пошук наступної резольвентної пари за допомогою однієї з наступних двох рекомендованих стратегій.

Перша стратегія має назву – стратегія від фактів. Якщо серед хорнівських диз'юнктів є факт, тобто одинарна додаткова літера, то треба шукати другий диз'юнкт, який складає з ним резольвентну пару. Після застосування правила резолюцій до такої пари отримується резольвента, яка буде коротше другого диз'юнкта

Друга стратегія має назву – стратегія від цілі. Якщо серед хорнівських диз'юнктів є цільовий, тобто диз'юнкт без додаткової літери, то треба шукати другий диз'юнкт, який з ним складає резольвентну пару. Далі кожного разу для наступної резольвенти потрібно шукати парний їй диз'юнкт та застосовувати правило резолюцій. Отримаємо під цілей до зупинки.

3.2. Приклад граматики реалізації методу резолюцій, доведення в логіці висловлень

Приклад 1. Використовуючи метод резолюцій дізнатись, чи є логічно правильним наступне міркування:

Архітектор вирішив працювати за комп'ютером (Z), або вирішив креслити на папері (X). Він не креслив на папері. Отже, інженер працював за комп'ютером.

Розв'язування. За пишемо дане міркування наступним чином: $Z \vee X, \bar{X}, Z$. Суперечність множини диз'юнктів доведемо за допомогою методу резолюцій.

Введемо позначення:

$$d_1 = Z \vee X, d_2 = \bar{X}, d_3 = \bar{Z},$$

Знайдемо резольвенту для $\{d_1, d_3\}$. Отримаємо новий диз'юнкт $d_4 = Z$. Завершуємо резольвентну пару $\{d_3, d_4\}$ буде $d_5 = \wedge$. На цьому можна вважати рішення задачі завершеним.

Приклад 2. Нехай $G = (a \vee b) \wedge (a \vee c) \wedge (\bar{b} \vee \bar{c}) \wedge \bar{a}$. Доведемо, що G – тотожно-хибна формула методом резолюції[3].

Розв'язування.

Введемо позначення: $d_1 = a \vee b$, $d_2 = a \vee c$, $d_3 = \bar{b} \vee \bar{c}$, $d_4 = \bar{a}$;

Знайдемо серед диз'юнктивів d_i резольвентну пару. Таких пар багато як приклад:

$$\{d_1, d_3\};$$

$$\{d_1, d_4\};$$

$$\{d_2, d_3\};$$

$$\{d_2, d_4\};$$

Знайдемо резольвенту для $\{d_1, d_3\}$. Отримаємо $d_5 = a \vee \bar{c}$. Далі застосовуємо правило резолюції для пари $\{d_2, d_5\}$. Отримаємо новий диз'юнкт $d_6 = a$. На завершення резольвентою для пари $\{d_4, d_6\}$ буде $d_7 = \wedge$. На цьому можна вважати рішення задачі завершеним.

Для зручності застосування методу резолюцій використовують наступний запис (таб. 2.2.1) :

1	$a \vee b$
2	$a \vee c$
3	$\bar{b} \vee \bar{c}$
4	$\bar{a}$
5	$a \vee \bar{c}$ для 1,3.
6	a для 2,5.
7	$\wedge$ для 4,6.

Рисунок 3.2.1 – допис до прикладу 2

Приклад 3. Розглянемо знову ту ж формулу G , що і у попередньому прикладі. Тільки зробимо заміну літери a на літеру $\bar{p}$, щоб всі диз'юнкти в КНФ для G були хорнівські.

$$G = (\bar{p} \vee \bar{b}) \wedge (\bar{p} \vee c) \wedge (\bar{b} \vee \bar{c}) \wedge p$$

Будемо доводити тотожну хибність формули G методом резолюцій, використовуючи стратегію від фактів. В такому випадку похідний факт один – диз'юнкт d_4 .

Введемо позначення та сформуємо його у вигляді наступного допису (таб. 2.2.2):

1	$\bar{p} \vee b$
2	$\bar{p} \vee c$
3	$\bar{b} \vee \bar{c}$
4	$\bar{p}$
5	b для 4,1
6	c для 4,2
7	$\bar{c}$ для 5,3
8	Λ для 6,7

Рисунок 3.2.2 – допис до прикладу 3

Тепер для порівняння для тієї ж задачі використаємо стратегію від цілі. Цільовий диз'юнкт буде d_3 . (таб. 2.2.3)

1	$\bar{p} \vee b$
2	$\bar{p} \vee c$
3	$\bar{b} \vee \bar{c}$
4	$\bar{p}$
5	$\bar{p} \vee c$ для 3,1
6	$\bar{p}$ для 5,2
7	Λ для 6,4

Рисунок 3.2.3 – допис до прикладу 3

3.3. Уніфікація. Приклад граматики реалізації та доведення в логіці предикатів

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.

Приклад 1. Перенесемо метод резолюцій у логіку предикатів. Основна ідея тут належить математику Жаку Арбрану.

Довести загальнозначущість формули логіки предикатів.

$$F = \left(\left((\forall x)(P(x) \rightarrow Q(x)) \right) \vee \left((\forall x)(Q(x) \rightarrow R(x)) \right) \right) \rightarrow \left((\forall x)(P(x) \rightarrow R(x)) \right).$$

Перейдемо до формули $G = \overline{F}$ та розробимо тісне заперечення в G :

$$G = \overline{F} = \overline{H_1 \wedge H_2 \rightarrow H_3} = H_1 \wedge H_2 \wedge \overline{H_3},$$

$$\text{де } H_1 = (\forall x)(P(x) \rightarrow Q(x)),$$

$$H_2 = (\forall x)(Q(x) \rightarrow R(x)),$$

$$H_3 = (\forall x)(P(x) \rightarrow R(x)).$$

$$\begin{aligned} \overline{H_3} &= \overline{(\forall x)(P(x) \rightarrow R(x))} = (\exists x) \overline{(P(x) \rightarrow R(x))} = (\exists x) \overline{(P(x) \vee \overline{R(x)})} = \\ &= (\exists x) \overline{(P(x) \wedge R(x))} = P(a) \wedge \overline{R(a)}. \end{aligned}$$

Зробимо заміну зв'язної «х» в «х» на «у», винесемо всі квантори загальності в G за дужки та представимо в H в КНФ:

$$G = (\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)}.$$

Можна застосувати метод резолюцій з уніфікацією, використовуючи стратегію від фактів.

1. $\overline{P(x)} \vee Q(x)$.
2. $\overline{Q(y)} \vee R(y)$.
3. $P(a)$.
4. $\overline{R(a)}$.
5. $Q(a)$ із 3, 1 при $x = a$.
6. $R(a)$ із 5, 2 при $y = a$.
7. $\perp$ із 6, 4.

3.4. Алгоритм роботи тренажера

Перед користувачем відображається головне вікно програми тренажера, у даному вікні знаходиться інформація, що до назви тренажера та є кнопка, що дозволяє переглянути теоретичний матеріал, яка має назву «Інформаційна сторінка», також знаходиться кнопка «Перейти до тестування». При натисненні на дану кнопку перед користувачем виводиться дві кнопки: «Метод резолюцій до логіки висловлень» та «Метод резолюцій до логіки предикатів». При виборі кнопки «Інформаційна сторінка», користувач має змогу ознайомитись із теоретичним матеріалом з теми «Метод резолюцій». При виборі кнопок «Перейти до тестових завдань по методу резолюцій», може обрати приклади рішення задач методом резолюцій.

Користувач може ознайомитись із завданням задач по мету резолюцій та обрати варіант відповіді. На кожному кроці виведено завдання, питання та варіанти відповіді, при цьому на кожному етапі буде кнопка «Інформаційна сторінка». Що дає змогу звернутись до теоретичного матеріалу, щоб подолати труднощі які можуть виникнути при проходженні тестів.

При виборі пункту «Метод резолюцій до логіки висловлень» користувач матиме змогу вирішити задачі методом резолюції у тестовому режимі. Тобто алгоритм рішення задач методом резолюції представлений як тестування із вибором відповіді. При обранні правильної відповіді користувач переходить до

наступного кроку рішення, якщо відповідь не вірна - виводиться повідомлення про помилку і відповідь не зараховується як правильна при повторній відповіді. За кожну правильну відповідь студент (користувач отримуватиме 1-бал)

Розглянемо усі кроки алгоритму тренажера.

3.4.1. Алгоритм застосування методу резолюцій до логіки висловлень.

Крок 1. Відображається питання: «Що у логіці висловлень називають літерою? Обрати правильне означення поняття літера та приклад»

Відображається три варіанти відповіді:

а) Елементарну формулу логіки висловлень, або її заперечення, називають літерою.

Приклад: $A \Rightarrow B, a \Rightarrow b = b \Leftarrow a$.

б) Елементарну формулу логіки висловлень, або її погодження, називають літерою.

Приклад: $A \Rightarrow B = \bar{Z}, a \Rightarrow \bar{Z}$

в) Елементарну формулу логіки висловлень, або її заперечення, називають літерою.

Приклад: $a = Z, a = \bar{Z}$.

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 2. Відображається питання: «Що називають диз'юнктом?»;

Відображається три варіанти відповіді:

а) Скінченна диз'юнкція літер, або їх заперечень;

б) Диз'юнкція літер протилежних значень, записаних у прямому вигляді;

в) Скінченна кон'юнкція літер, або їх заперечень;

Якщо вибрано правильну відповідь, то відбувається перехід на наступний крок. Якщо ні – виводиться повідомлення про помилку, з'являється інформативний напис про неправильність відповіді, та підказка:

«Скінченна диз'юнктив літер, або їх заперечення називають диз'юнктами.

Приклад: $a \vee b$;»

Студент робить відповідні висновки і вибирає правильну відповідь.

Крок 3. Відображається питання: «Оберіть правильне означення поняття - «Резольвентна пара»:».

Відображається три варіанти відповіді:

а) Два кон'юнкти називаються резольвентною парою, якщо існує така тотожність, яка їх об'єднує.

б) Два диз'юнкта називаються резольвентною парою, якщо існує така літера, яка бере участь в одному диз'юнкті як додатна (записана в прямому вигляді), а в другому – як від'ємна (записана у інверсному вигляді).

в) Два диз'юнкта називаються резольвентною парою, якщо існує така літера, яка бере участь в двох диз'юнктах як від'ємна.

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 4. Задано резольвентну пару $a \vee b \vee c$ та $\bar{c} \vee d \vee f$. Знайти резольвенту.

а) $a \vee b \vee d \vee f$

б) $c \vee b \vee d \vee f$

в) $a \vee c \vee b \vee d$

Якщо вибрано правильну відповідь, то відбувається перехід на наступний крок. Якщо ні – виводиться повідомлення про помилку, з'являється інформативний напис про неправильність відповіді, та підказка:

«Теорема: Нехай d_1 та d_2 – резольвентна пара диз'юнктив вигляду $d_1 = a \vee d$, $d_2 = c \vee \bar{d}$ через a, c позначимо члени диз'юнктив з невиділеними літерами. Тоді отримаємо $(a \vee d) \wedge (c \vee \bar{d}) \Rightarrow (a \vee c)$;»

Студент робить відповідні висновки і вибирає правильну відповідь

Крок 5. Відображається питання: «Кон'юнктивна нормальна форма (КНФ) в булевій логіці – :»

Відображається три варіанти відповіді:

а) ... нормальна форма, в якій булева формула має вид диз'юнкції декількох кон'юнкцій;

б) ... нормальна форма, в якій булева формула має вид кон'юнкції декількох диз'юнктив;

в) ... нормальна форма, в якій булева формула має вид резольвенти одного диз'юнкту.

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 6. Відображається умова: «Довести методом резолюцій, що формула F – тавтологія (загальнозначуща). Нехай $\bar{F} = G = (x \vee z) \wedge (x \vee c) \wedge (\bar{z} \vee \bar{c}) \wedge \bar{x}$.

Умова задачі відображається на кожному наступному кроці алгоритму.

Необхідно довести, що $F = \square$, а $G = \square$.

Відображається три варіанти відповіді:

а) $F=0, G=1$;

б) $F=1, G=0$;

в) $F=1, G=1$;

Якщо значення задані не вірно виводиться повідомлення про помилку з наступним повідомленням:

В текстовому повідомленні пояснення: «За умовою необхідно довести, що F – тавтологія, тобто $F = 1$, а так як $\bar{F} = G$, то $G = 0$.»

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 7. Відображається умова: «У формулі G є чотири диз'юнкта d_1, d_2, d_3, d_4 , які рівні відповідно:»

Відображається три варіанти відповіді:

а) $d_1 = (x \vee z)(x \vee c), d_2 = (x \vee c)(\bar{z} \vee \bar{c}), d_3 = (\bar{z} \vee \bar{c})\bar{x}, d_4 = \bar{x};$

б) $d_1 = x \vee \bar{z}, d_2 = \bar{z} \vee c, d_3 = \bar{z} \vee \bar{c}, d_4 = x;$

в) $d_1 = x \vee z, d_2 = x \vee c, d_3 = \bar{z} \vee \bar{c}, d_4 = \bar{x};$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Після даного кроку, для зручності застосування методу резолюції, до умови буде додано наступний допис:

1	$x \vee z$
2	$x \vee c$
3	$\bar{z} \vee \bar{c}$
4	$\bar{x}$

Крок 8. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктивів відбуватиметься пошук резольвентної пари?»

Відображається три варіанти відповіді:

а) $\{d_2, d_3\}; \{d_2, d_1\};$

б) $\{d_1, d_2\}; \{d_3, d_4\};$

в) $\{d_1, d_3\}; \{d_2, d_3\};$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 9. Відображається умова: «Нехай обрано резольвентну пару $\{d_1, d_3\}$. Знайти серед наступних диз'юнктивів резольвенту пари $\{d_1, d_3\}$ (застосовуючи правило резолюції), щоб отримати d_5 :»;

Відображається три варіанти відповіді:

а) $d_5 = x \vee \bar{c}$;

б) $d_5 = z \vee \bar{c}$;

в) $d_5 = x \vee \bar{z}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Після даного кроку, для зручності застосування методу резолюції, до умови буде додано наступний допис:

1	$x \vee z$
2	$x \vee c$
3	$\bar{z} \vee \bar{c}$
4	$\bar{x}$
5	$x \vee \bar{c}$

Крок 10. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктив відбудуватиметься пошук резольвентної пари?»

Відображається три варіанти відповіді:

а) $\{d_2, d_3\}$;

б) $\{d_2, d_5\}$;

в) $\{d_1, d_5\}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 11. Відображається умова: «Нехай обрано резольвентну пару $\{d_2, d_5\}$. Знайти серед наступних диз'юнктив резольвенту пари $\{d_2, d_5\}$ (застосовуючи правило резолюції), щоб отримати d_6 :»;

Відображається три варіанти відповіді:

а) Отримаємо: c ;

б) Отримаємо: x ;

в) Отримаємо: $\bar{x}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Після даного кроку, для зручності застосування методу резолюції, до умови буде додано наступний допис:

1	$x \vee z$
2	$x \vee c$
3	$\bar{z} \vee \bar{c}$
4	$\bar{x}$
5	$x \vee \bar{c}$
6	x

Крок 12. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктив відбуватиметься пошук резольвентної пари?

Відображається три варіанти відповіді:

а) $\{d_3, d_4\}$;

б) $\{d_4, d_5\}$;

в) $\{d_4, d_6\}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 13. Відображається питання: «Яким символом позначається порожній диз'юнкт?»;

Відображається три варіанти відповіді:

а) $\vee$;

б) $\wedge$;

в) $\perp$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 14. Відображається умова: «Знайдіть серед диз'юнктив d_i резольвентну пару для $\{d_4, d_6\}$ (застосовуючи правило резолюції), щоб отримати $\{d_7\}$ »;

Відображається три варіанти відповіді:

а) Отримаємо: 0;

б) Отримаємо: 1;

в) Отримаємо: x ;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 15. Виводиться повідомлення про закінчення розв'язку та кінцевий результат та висновок:

«Так як отримано порожній диз'юнкт, то формула $G = 0$, а тому $F = \overline{G} = 1$, тобто формула F є тавтологією»

Пропонується пройти розв'язок знову або завершити його. Якщо вибрано повторне проходження, то відбувається перехід на крок 0. Також можна перейти до тестування іншої задачі, або теоретичного тестування. Можна повторно ознайомитись з теоретичною інформацією обравши пункт «Інформаційна сторінка».

3.4.2. Алгоритм застосування методу резолюцій до логіки предикатів.

Крок 1. Відображається питання: «Оберіть визначення яке най більше підходить для логіки предикатів»

Відображається три варіанти відповіді:

а) Логіка 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.;

б) Логіка предикатів — це дедуктивна теорія, яка моделює процес виведення одних висловлювань із інших, враховуючи їх структуру;

в) Логіка предикатів — позначає висловлювання про певний непорожній клас, в якому властивість P притаманна лише деяким елементам цього класу, тобто існують елементи класу A , яким притаманна властивість P ;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 2. Відображається питання: «Як називається окремо взятий предикат?»

Відображається три варіанти відповіді:

а) Елементарною формулою;

б) Диз'юнктом;

в) Квантором;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 3. Відображається умова: «Ввівши предикати формалізувати велику теорему Ферма: «Для будь-якого цілого $n > 2$ не існує натуральних чисел x, y, z , що задовольняє рівняння $x^n + y^n = z^n$ »

Введемо предикати:

Відображається три варіанти відповіді:

а) $p(x) = xN; q(n) = "n > 2", n; (x, y, z, n) = "x^n + y^n = z^n", (x, y, z, n);$

б) $p(x) = xN; q(n) = "n > 2", nN; r(x, y, z, n) = "x^n + y^n = z^n", (x, y, z, n)N * N * N * N;$

в) $p(x) = xN; q(n) = "n > 2", N; r(x, y, z) = "x^n + y^n = z^n", (x, y, z)N * N * N;$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 4. Відображається умова: «Після введення предикатів $p(x) = xN; q(n) = "n > 2", nN; r(x, y, z, n) = "x^n + y^n = z^n", (x, y, z, n)N * N * N * N;$

Теорема ферма має вигляд:»

Відображається три варіанти відповіді:

а) $x y z n[(p(x)p(y)p(z)q(n)) \rightarrow r(z * y * z * n)];$

б) $x y z n[(p(x)p(y)p(z)p(n)) \rightarrow r(z, y, z, n)];$

в) $x y z n[(p(x)p(y)p(z)q(n)) \rightarrow r(z, y, z, n)];$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 5. Виводиться умова: Довести методом резолюцій, що формула F — тавтологія (загальнозначуща) формули логіки предикатів. Нехай
$$F = \left(\left((\forall x)(P(x) \rightarrow Q(x)) \right) \vee \left((\forall x)(Q(x) \rightarrow R(x)) \right) \right) \rightarrow \left((\forall x)(P(x) \rightarrow R(x)) \right).$$

Умова задачі відображається на кожному наступному кроці алгоритму.

Перейдемо до формули $G = \overline{F}$ та розробимо тісне заперечення в G :

Відображається три варіанти відповіді:

а) $G = \overline{F} = \overline{H_1 \wedge H_2 \rightarrow H_3} = H_1 \wedge H_2 \wedge \overline{H_3};$

б) $G = \overline{F} = H_1 \wedge H_2 \wedge \overline{H_3}; = \overline{H_1 \wedge H_2 \rightarrow H_3};$

в) $G = \overline{F} = H_1 \wedge H_2 \wedge \overline{H_3}; = H_1 \wedge H_2 \rightarrow \overline{H_3};$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 6. Виводиться умова: «Згідно формули $G=\overline{F}$ тісне заперечення матиме вигляд: « $G = \overline{F} = \overline{H_1 \wedge H_2 \rightarrow H_3} = H_1 \wedge H_2 \wedge \overline{H_3}$;». Де $H_1, H_2, H_3 = \dots$:»

Відображається три варіанти відповіді:

а) $H_1 = (\forall x)(P(x) \rightarrow Q(x)); H_2 = (\forall x)(Q(x) \rightarrow R(x)); H_3 = (\forall x)(P(x) \rightarrow R(x))$

б) $H_1 = (P(x) \rightarrow Q(x)); H_2 = (Q(x) \rightarrow R(x)); \overline{H_3} = (P(x) \rightarrow R(x));$

в) $H_1 = (\forall x) \wedge (P(x) \rightarrow Q(x)); H_2 = (\forall x) \wedge (Q(x) \rightarrow R(x));$

$H_3 = (\forall x) \wedge (P(x) \rightarrow R(x));$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 7. Виводиться умова: «Згідно формули $G=\overline{F}$ тісне заперечення матиме вигляд: « $G = \overline{F} = \overline{H_1 \wedge H_2 \rightarrow H_3} = H_1 \wedge H_2 \wedge \overline{H_3}$;». Де $\overline{H_3} = \dots$:»

Відображається три варіанти відповіді:

а) $\overline{H_3} = \overline{(\forall x)(P(x) \rightarrow R(x))} = (\forall x)\overline{(P(x) \rightarrow R(x))} = (\forall x)\overline{(P(x) \vee R(x))} = (\forall x)(\overline{P(x) \wedge R(x)}) = P(a) \wedge \overline{R(a)}.$

б) $\overline{H_3} = \overline{(\forall x)(P(x) \rightarrow R(x))} = (\exists x)\overline{(P(x) \rightarrow R(x))} = (\exists x)\overline{(P(x) \vee R(x))} = (\exists x)(\overline{P(x) \wedge R(x)}) = P(a) \wedge \overline{R(a)}.$

в) $\overline{H_3} = \overline{(P(x) \rightarrow R(x))} = \overline{(P(x) \rightarrow R(x))} = \overline{(P(x) \vee R(x))} = (P(x) \wedge \overline{R(x)}) = P(a) \wedge \overline{R(a)}.$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 8. Виводиться умова: «Зробимо заміну зв'язної «х» в «х» на «у», винесемо всі квантори загальності в G за дужки та представимо в H в КНФ:»

Відображається три варіанти відповіді:

а) $G = \left(\overline{(P(x) \vee Q(x))} = \overline{(Q(y) \vee R(y))} \right) \wedge P(a) \wedge \overline{R(a)};$

$$\underline{\text{б)}} G=(\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)};$$

$$\text{в)} G=(\forall x)(\forall y) \left((P(x) \rightarrow Q(x)) \wedge (Q(y) \rightarrow R(y)) \right) \wedge P(a) \wedge \overline{R(a)};$$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 9. Виводиться умова: «Застосуємо метод резолюцій з уніфікацією, використовуючи стратегію від фактів до формули:

« $G=(\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)}$ ». У формулі G є чотири диз'юнкта d_1, d_2, d_3, d_4 , які рівні відповідно:»

Відображається три варіанти відповіді:

$$\text{а)} d_1 = \overline{P(x)} \vee Q(x); d_2 = \overline{Q(y)} \vee R(y); d_3 = P(a); d_4 = \overline{R(a)};$$

$$\text{б)} d_1 = \overline{P(x)}; d_2 = \overline{Q(y)}; d_3 = P(a); d_4 = \overline{R(a)};$$

$$\text{в)} d_1 = Q(x); d_2 = R(y); d_3 = P(a); d_4 = \overline{R(a)};$$

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Після даного кроку, для зручності застосування методу резолюції, до умови буде додано наступний допис:

$$G=(\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)}.$$

1	$\overline{P(x)} \vee Q(x).$
2	$\overline{Q(y)} \vee R(y).$
3	$P(a).$
4	$\overline{R(a)}$

Крок 10. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктив відбуватиметься пошук резольвентної пари?»

Відображається три варіанти відповіді:

а) $\{d_2, d_3\}$;

б) $\{d_3, d_4\}$;

в) $\{d_3, d_1\}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 11. Відображається умова: «Нехай обрано резольвентну пару $\{d_3, d_1\}$. Знайти серед наступних диз'юнктив резольвенту пари $\{d_3, d_1\}$ (застосовуючи правило резолюції), щоб отримати d_5 :»;

Відображається три варіанти відповіді:

а) $d_5 = Q(a)$ при $x = a$;

б) $d_5 = Q(x)$ при $x = 0$;

в) $d_5 = Q(a^x)$ при $x = 1$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

$$G = (\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)}.$$

1	$\overline{P(x)} \vee Q(x).$
2	$\overline{Q(y)} \vee R(y).$
3	$P(a).$
4	$\overline{R(a)}$
5	$Q(a) \text{ при } x = a$

Крок 12. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктив відбуватиметься пошук резольвентної пари?»

Відображається три варіанти відповіді:

а) $\{d_2, d_3\}$;

б) $\{d_5, d_2\}$;

в) $\{d_5, d_1\}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 13. Відображається умова: «Нехай обрано резольвентну пару $\{d_5, d_2\}$. Знайти серед наступних диз'юнктив резольвенту пари $\{d_5, d_2\}$ (застосовуючи правило резолюції), щоб отримати d_6 :»;

Відображається три варіанти відповіді:

а) Отримаємо: L ;

б) Отримаємо: $R(a)$ при $y = a$;

в) Отримаємо: $\overline{R(a)}$ при $x = a$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Після даного кроку, для зручності застосування методу резолюції, до умови буде додано наступний допис:

$$G = (\forall x)(\forall y) \left(\left(\overline{P(x)} \vee (Q(x) \wedge \overline{Q(y)} \vee R(y)) \right) \right) \wedge P(a) \wedge \overline{R(a)}.$$

1	$\overline{P(x)} \vee Q(x).$
2	$\overline{Q(y)} \vee R(y).$
3	$P(a).$
4	$\overline{R(a)}$
5	$Q(a)$ при $x = a$
6	$R(a)$ при $x = a$;

Крок 14. Відображається умова: «Вибрати всі резольвентні пари.

Згідно умови, до яких диз'юнктив відбудуватиметься пошук резольвентної пари?

Відображається три варіанти відповіді:

а) $\{d_3, d_4\}$;

б) $\{d_4, d_5\}$;

в) $\{d_6, d_4\}$;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 15. Відображається умова: «Знайдіть серед диз'юнктив d_i резольвентну пару для $\{d_6, d_4\}$ (застосовуючи правило резолюції), щоб отримати $\{d_7\}$:»;

Відображається три варіанти відповіді:

а) Отримаємо: a ;

б) Отримаємо: A ;

в) Отримаємо: F ;

Якщо вибрано правильну відповідь, то відбувається перехід на наступне питання. Якщо відповідь не вірна, з'являється повідомлення про неправильність відповіді. Користувач аналізує свою відповідь та відповідає повторно.

Крок 16. Виводиться повідомлення про закінчення розв'язку та кінцевий результат та висновок:

«Так як отримано порожній диз'юнкт, то формула $G = 0$, а тому $F = \overline{G} = 1$, тобто формула F є тавтологією»

Пропонується пройти розв'язок знову або завершити його. Якщо вибрано повторне проходження, то відбувається перехід на крок 1. Також можна перейти до тестування іншої задачі, або теоретичного тестування. Можна повторно ознайомитись з теоретичною інформацією обравши пункт «Інформаційна сторінка».

3.5. Блок-схема алгоритму роботи тренажера

На зображеннях 3.5.1 – 3.5.5 зображена блок-схема алгоритму роботи програми-тренажера.

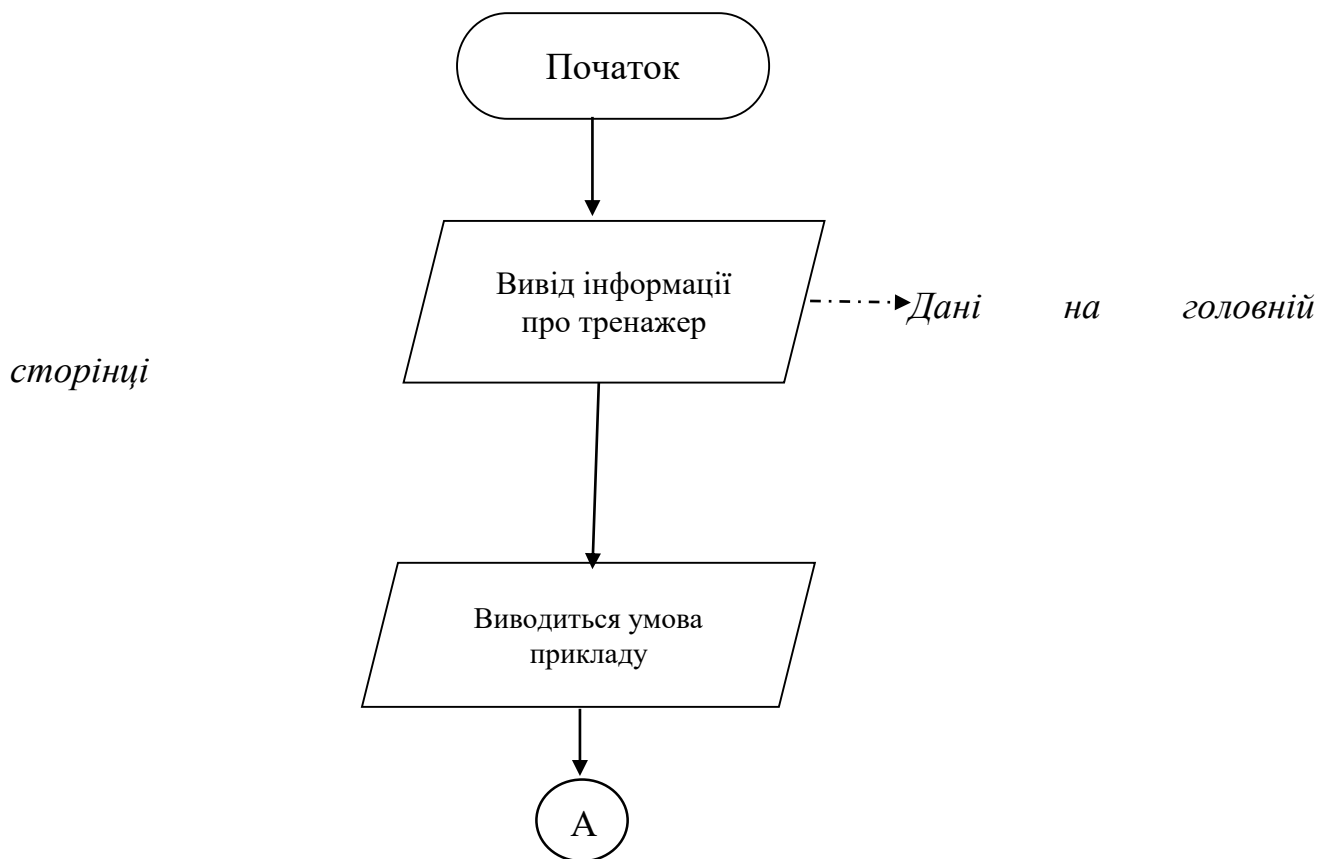


Рисунок 2.5.1 – Блок-схема головної сторінки тренажера

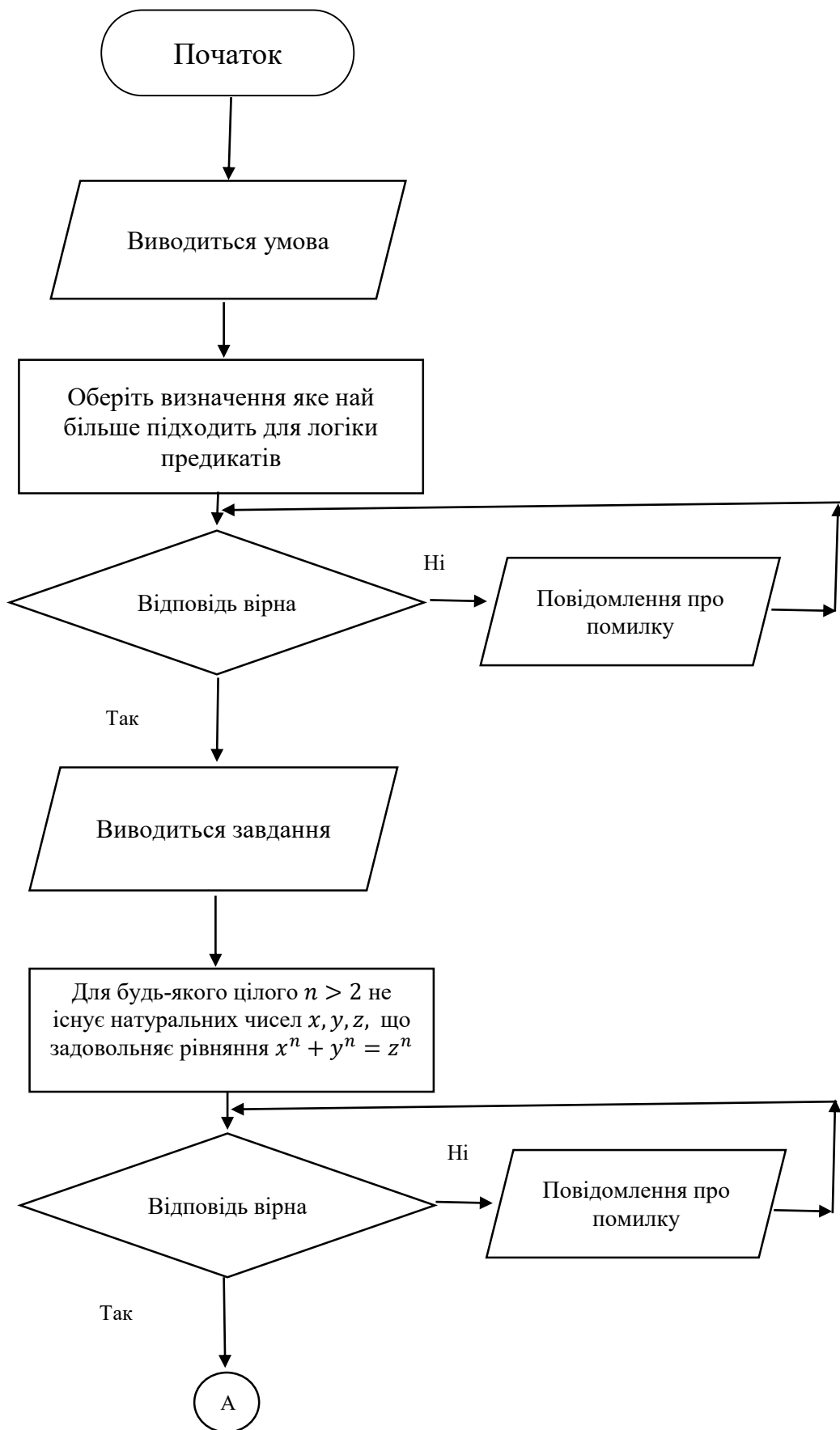


Рисунок 3.5.2 Блок схема «Розв'язок» для прикладів

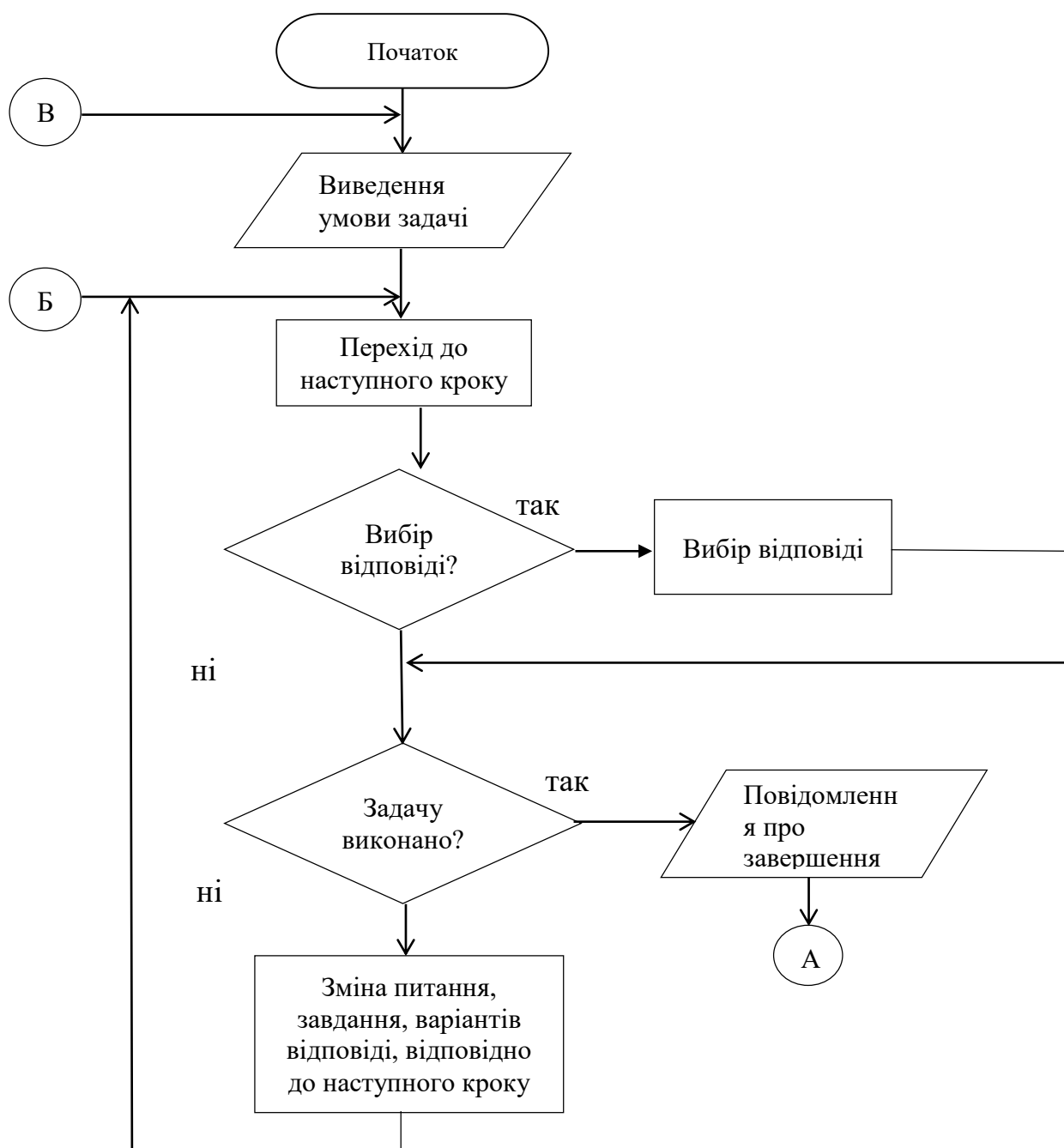


Рисунок 3.5.3 – Блок-схема алгоритму роботи тренажера

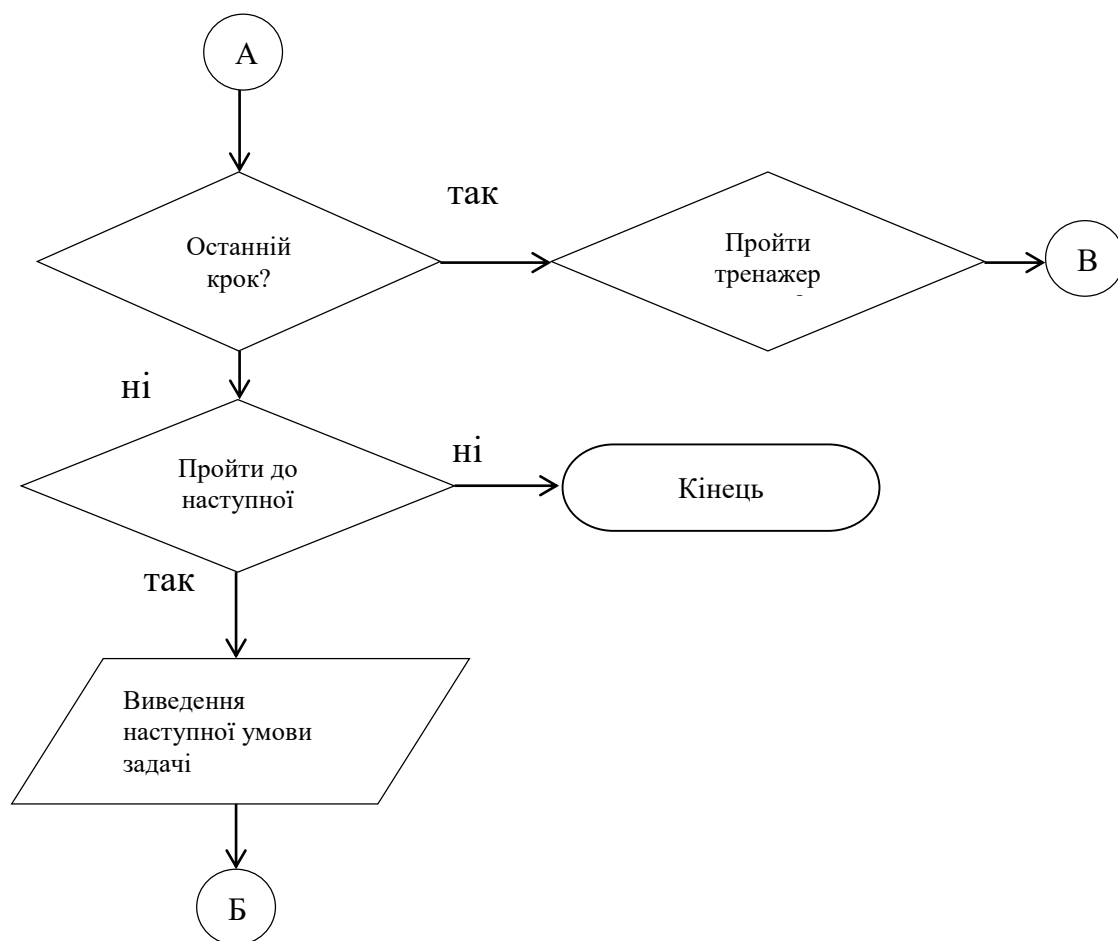


Рисунок 3.5.4 – Продовження блок-схеми останнього кроку тренажера

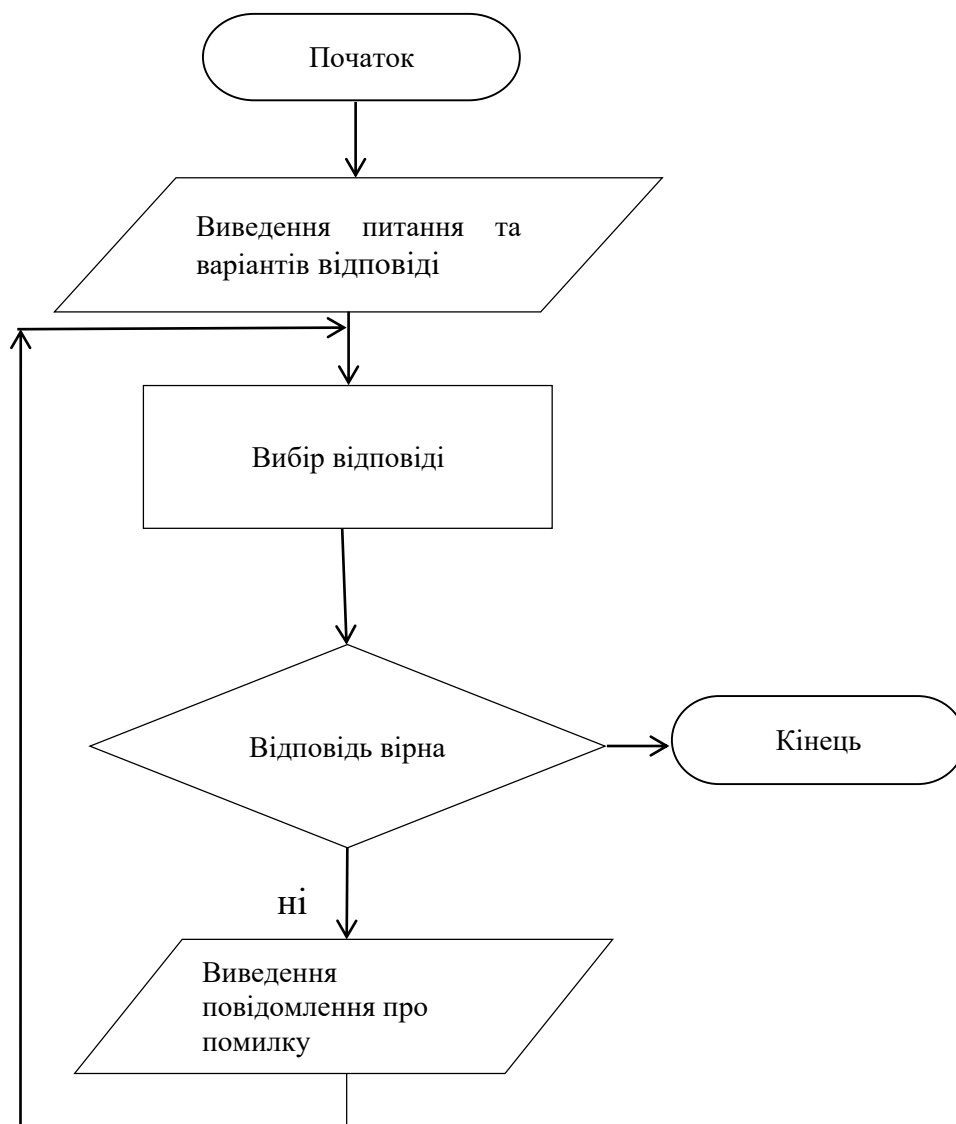


Рисунок 3.5.5 – Блок-схема процесу «Вибір відповіді»

4. ПРАКТИЧНА ЧАСТИНА

4.1. Обґрунтування вибору програмних засобів

Нові мови програмування з'являються щороку. А основні затребувані розширюються і прогресують. Оскільки мова програмування C# була створена і супроводжується компанією Microsoft, тому до даного продукту періодично робить масштабні модернізації з додавання корисних функції до C#. Також на C# (C Sharp) написано вже мільярди рядків коду по цілому світу, не тільки під Microsoft, отже роботи передбачається вдосталь усім бажаючим вивчати мову програмування C# (C Sharp)

C# має великий набір функцій для використання, що дозволяє використовувати його у багатьох випадках, в тому числі і для створення навчального тренажеру.

Що стосується порівняння мов програмування, слід відзначити, мова програмування C# (C Sharp) високорівнева, це означає, що вона дещо схожа на англійську. Мова програмування C# має строгу статичну типізацію, підтримує поліморфізм, перевантаження операторів, вказівники на функції-члени класів, атрибути, події, властивості, винятки, коментарі у форматі XML. Переїнявши багато що від своїх попередників — мов C++, Delphi, Модула і Smalltalk — у C#, опираючись на практику їхнього використання, навмисне виключили деякі моделі, що зарекомендували себе як проблематичні при розробці програмних систем у вище перелічених мовах програмування[10].

C# керується чіткими суровими правилами використання пріоритетів, а значить абстрагує програмістів початківців робити дурні помилки і навчатися її, смаживши свій мозок.

Синтаксис досить мінімалістичний C#. З ручним управлінням пам'яттю. Багатьом вказана обставина незручність, проте слідкування за правильністю

функцій, розуміння передачі аргументів тісно пов'язане вивчення мови програмування C# (C Sharp).

Оскільки синтаксис C# близький до C, C++ і Java, то, вільно володіючи C#, можна оволодіти і іншими мовами[11].

4.2. Опис процесу програмної реалізації

Навчальний тренажер створений на мові програмування C# (.NET Framework 2.0).

У тренажері виведено окремий клас, який підключає тести, питання та відповіді з окремих ресурсів. Тобто вся інформація відокремлена від коду та виведена у папку «Resurs». Тому для коректної роботи програми повинна бути біля вікна програми папка «Resurs».

Тепер пройдемо про пунктам та класам програми:

«Data» - окремий клас, який формує шлях до файлу з питаннями:

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.IO;

namespace TestRobAleksov
{
    public class Data
    {
        // public string[] answers;
        public static string path = "Resurs/Questions/";

        public string path0;
        public string path1;
        public string path2;
        public string path3;
        public string path4 = "";

        public Data(int i)
        {
            path0 = path + "Ques" + i + "0.rtf";
            path1 = path + "Answ" + i + "1.rtf";
            path2 = path + "Answ" + i + "2.rtf";
            path3 = path + "Answ" + i + "3.rtf";
            path4 = path + "Help" + i + "0.rtf";
        }
    }
}
```

Як можна помітити тут присутні п'ять полів, path0 - це вивід (шлях) до самого питання тренажеру. Path1-3 - варіанти відповіді (а)б)в)). path4 - довідкова інформація при неправильній відповіді.

«Form1» – основна форма з якої розпочинається робота (Додаток А)

«Programs» - основний клас з якого запускається програма:

```
namespace TestRobAleksov
{
    static class Program
    {
        /// <summary>
        /// Главная точка входа для приложения.
        /// </summary>
        [STAThread]

        static void Main()
        {
            Application.EnableVisualStyles();
            Application.SetCompatibleTextRenderingDefault(false);
            Application.Run(new Form1());
        }
    }
}
```

«PractT1» - вікно вибору теми (Додаток Б)

«AnswRead »– виведення невірної довідки (Додаток В)

«MessadgEnd» – вікно з кінцевим результатом та варіантами подальшого вибору (Додаток Г)

4.3. Інструкція по використанню програми-тренажеру

Тренажер призначений для удосконалення знань, теми «Метод резолюцій», яка є розділом математичної логіки.

Запускається тренажер із ярлика програми тренажеру (рис 1):

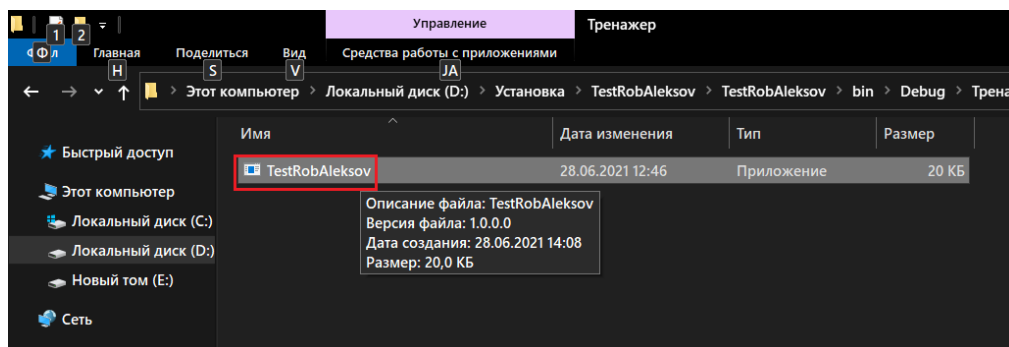


Рисунок 4.3.1 – Ярлик запуску програмного тренажеру з теми: «Метод резолюцій».

Після запуску тренажера відображається стартове вікно з інформацією (рис. 2.):

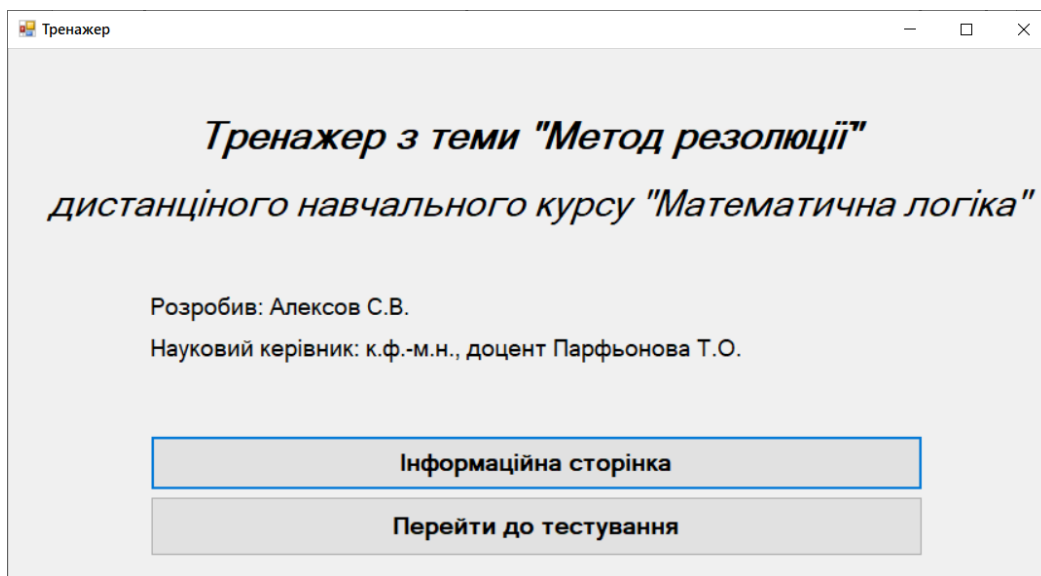


Рисунок 4.3.2 – Стартове вікно тренажера

Щоб ознайомитися з теоретичним матеріалом необхідно натиснути кнопку «Інформаційна сторінка». Після чого з'явиться вікно з теоретичним матеріалів (рис. 3). Даному вікні також присутня кнопка повернення на головну сторінку.

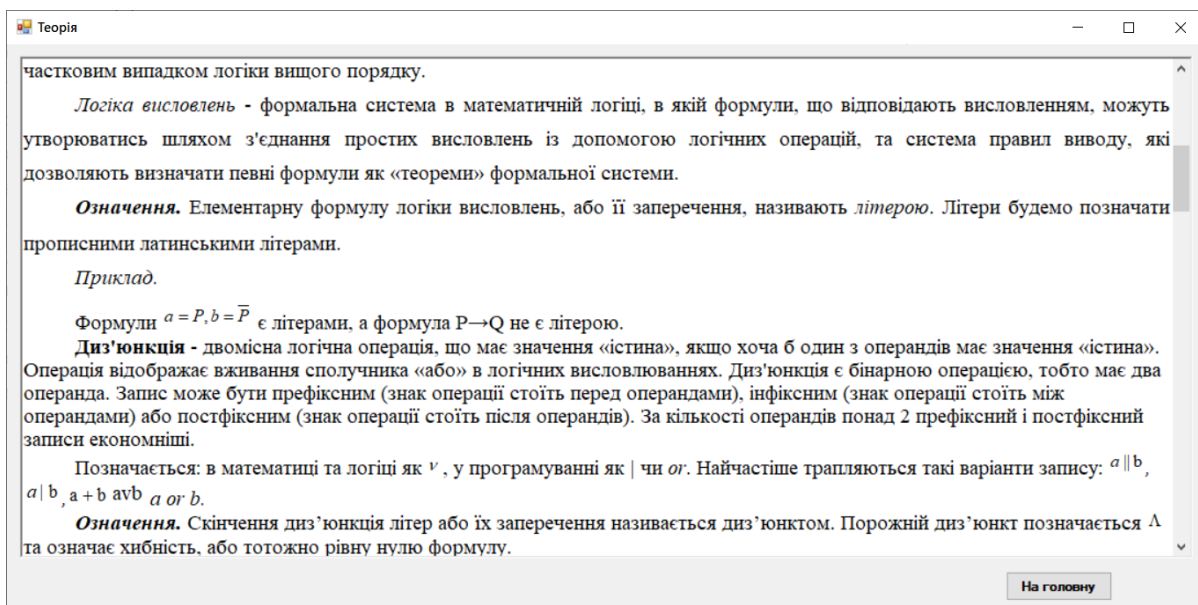


Рисунок 4.3.3 – Інформаційна сторінка тренажеру з теми «Методу резолюцій»

Натиснувши кнопку «Перейти до тестування», перед користувачем відкривається вікно де він може обрати тестування: «Метод резолюцій до логіки висловлень», або «Метод резолюцій до логіки предикатів» (рис 4).

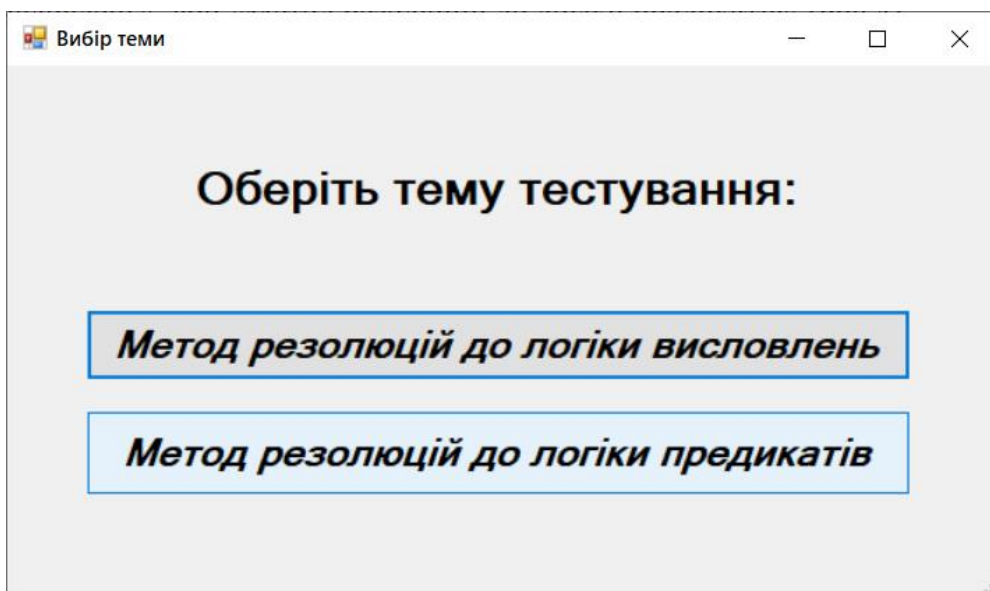


Рисунок 4.3.4 – Вибір теми тестування

При натисканні на кнопку «Метод резолюцій до логіки висловлень». Користувач проходить тестування по методу резолюцій застосованому виключно до логіки висловлень (рис.5)

Тестування

Довести методом резолюцій, що формула F - тавтологія (загальнозначуща).
Нехай $\bar{F} = G = (x \vee z) \wedge (x \vee c) \wedge (\bar{z} \vee \bar{c}) \wedge \bar{x}$.

1	$x \vee z$
2	$x \vee c$
3	$\bar{z} \vee \bar{c}$
4	$\bar{x}$

Вибрати всі резольвентні пари.
Згідно умови, до яких диз'юнктив відбуватиметься пошук резольвентної пари?

Рисунок 4.3.5 – Приклад одного із питань тренажера

При невірній відповіді виводиться наступне повідомлення (рис 6):

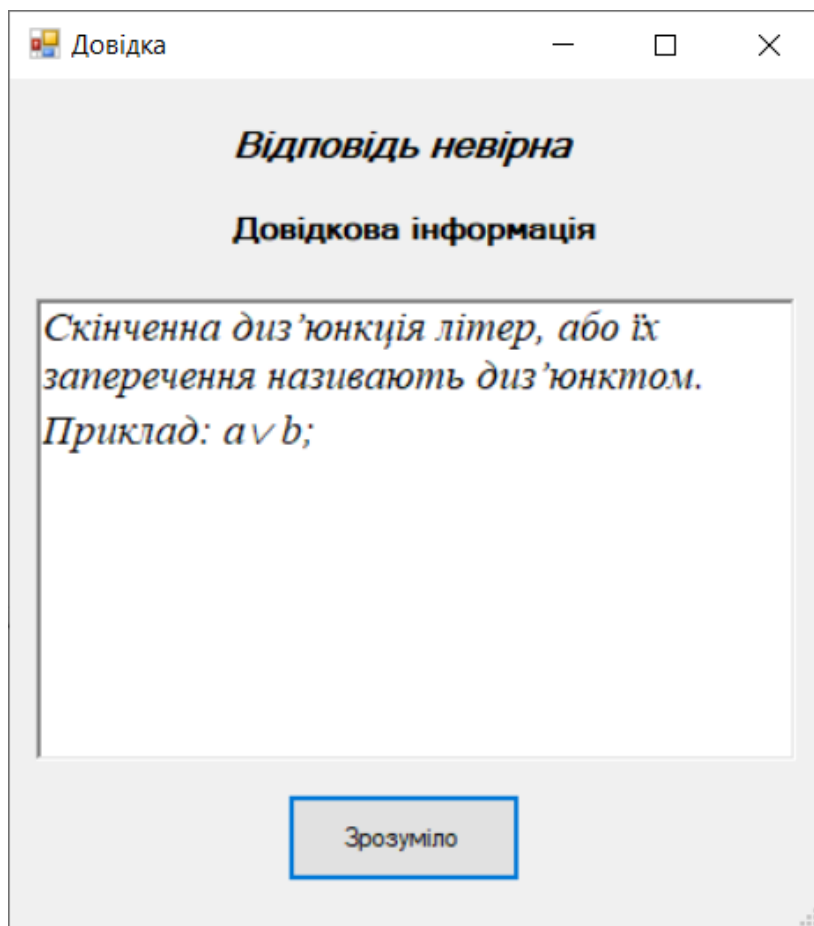


Рисунок 4.3.6 – Повідомлення про помилку та підказка

Після проходження тренажеру виводиться результат тестування та пропозиція по наступним дії користувача. (рис 7):

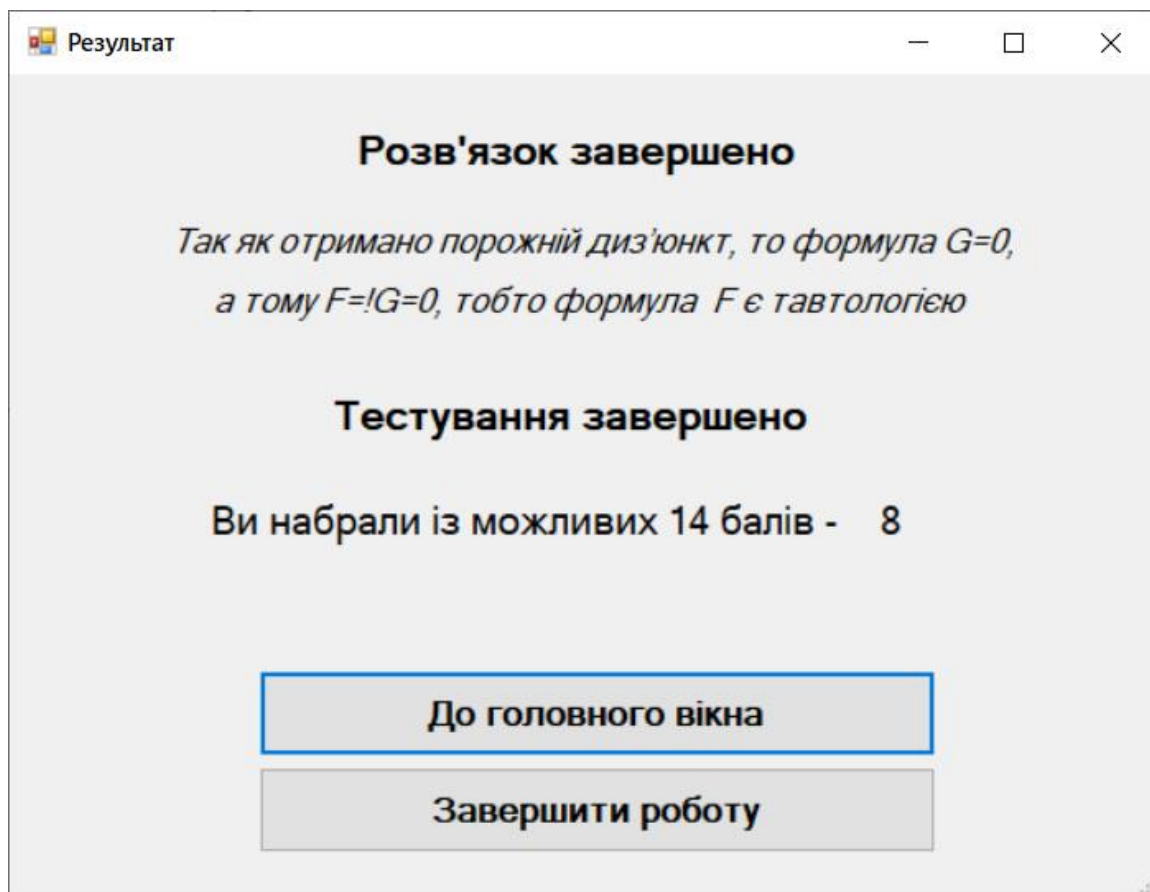


Рисунок 4.3.7 – Повідомлення про завершення тестування, та довідка про результат тестування

Студент (користувач) може повторно пройти тестування, ознайомитись із теоретичними відомостями, або пройти тестування з іншої теми.

ВИСНОВКИ

Перед створенням алгоритму та написанням тренажеру було розроблено велику кількість дій, які були спрямовані на формування матеріалу та розроблення задач.

Усі поставлені завдання були виконані.

Результатом успішного виконання поставлених завдань є тренажер з теми «Метод резолюцій» дистанційного навчального курсу «Математична логіка»

Для досягнення мети були виконані наступні завдання:

- розглянуто теоретичні відомості та специфіку застосування методу резолюцій;
- складено алгоритм тренажера з теми «Метод резолюцій» дистанційного навчального курсу «Математична логіка»;
- складено блок-схему алгоритму;
- розроблено програмну реалізацію тренажеру.
- розписано інструкція по використанню тренажера

Робота виконана з наведеними актуальністю обраної теми, описом програмної реалізації та інструкцією по використанню програми.

Для реалізації тренажера була використана мова програмування C# у середовищі розробки Visual Studio.

Було розроблено два приклади по темі «Метод резолюцій», та успішно їх включено у програму тренажер. В алгоритмах покроково розглянуто рішення задач, алгоритм склав основу програмної реалізації. Також сформовано інформаційну сторінку, яка слугує як підказка до рішення задач тренажеру.

Даний тренажер може використовуватися як навчальний у дистанційному курсі з дисципліни «Математична логіка».

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Ємець О. О. Методичні рекомендації щодо оформлення пояснювальних записок до курсових проектів (робіт) для студентів напряму підготовки 6.040302 «Інформатика» та спеціальності 7.04030203, 8.04030203 «Соціальна інформатика» ПУЕТ / О. О. Ємець, Ол-ра О. Ємець. – Полтава : ПУЕТ, 2013. – 60 с.
2. Нікольський Юрій Володимирович. Дискретна математика/ Ю.В. Нікольський, В.В. Пасічник, Ю.М. Щербина – Київ: Видавнича група ВНУ, 2007. – 368 с.: іл. – ISBN 966-552-201-9
3. Матвієнко Микола. Математична логіка та теорія алгоритмі: навчальний посібник/ Матвієнко М., Шаповалов С. – К.: Видавництво Ліра-К, 2015. – 212 с.: іл. – SBN 978-966-2609-74-5
4. Козлакова Г. О. Використання засобів Інтернет у науково-педагогічних дослідженнях. Нові інформаційні технології в начальних закладах України / Г. О. Козлакова // Матеріали Міжнародної конференції пам'яті проф. І. І. Мархеля . – Одеса, 2005.
5. Кравець В.О. Меморандум створення інформаційної освітньої мережі "Українська дистанційна освіта" / В.О. Кравець, В.М. Кухаренко . – Режим доступу: <http://www.osvita.org.ua/distance/ukraine/add/00/>
6. Джон Д. Келлехер, Основы машинного обучения для аналитического прогнозирования: алгоритмы, рабочие примеры и тематические исследования., 1-е изд. : Пер. с англ. / Джон Д. Келлехер, Брайан Мак-Нейми, Аоифе д'Арси. – М. : «Діалекс-Вильямс», 2019. – 656 с.
7. Бібліографічний запис. Бібліографічний опис. Загальні вимоги та правила складання: ДСТУ 7.1-2006. – [Чинний від 2007-07-01]. – К. : Держспоживстандарт України, 2007. – 47 с.
8. А. Якобсон. Унифицированный процесс разработки программного обеспечения. А. Якобсон, Г. Буч, Дж. Рамбо - 492 с

9. В.М.Ткачук Алгоритми і структура даних: Навчальний посібник / В.М.Ткачук. - ІваноФранківськ : Видавництво Прикарпатського національного університету імені Василя Стефаника, 2016. 286 с.
10. Глоба Л. С. Розробка інформаційних ресурсів та систем [Електронний ресурс] : конспект лекцій / Л. С. Глоба, Т. М. Кот. - Київ : НТУУ "КПІ", 2014. - 318 с.
11. Давыдов В.Г. Программирование и основы алгоритмизации: Учеб. пособие. / В.Г. Давыдов. М.: Высш. шк., 2003. 447 с.
12. Энциклопедия языков программирования : [URL]. - Режим доступа: <http://progopedia.ru/>

Додаток А.

Програмний код основної форми

```

partial class Form1
{
    /// <summary>
    /// Об'язательная переменная конструктора.
    /// </summary>
    private System.ComponentModel.IContainer components = null;

    /// <summary>
    /// Освободить все используемые ресурсы.
    /// </summary>
    /// <param name="disposing">истинно, если управляемый ресурс должен быть удален;
    иначе ложно.</param>
    protected override void Dispose(bool disposing)
    {
        if (disposing && (components != null))
        {
            components.Dispose();
        }
        base.Dispose(disposing);
    }

    #region Код, автоматически созданный конструктором форм Windows

    /// <summary>
    /// Требуемый метод для поддержки конструктора — не изменяйте
    /// содержимое этого метода с помощью редактора кода.
    /// </summary>
    private void InitializeComponent()
    {
        this.label1 = new System.Windows.Forms.Label();
        this.label2 = new System.Windows.Forms.Label();
        this.button1 = new System.Windows.Forms.Button();
        this.button2 = new System.Windows.Forms.Button();
        this.label3 = new System.Windows.Forms.Label();
        this.label4 = new System.Windows.Forms.Label();
        this.SuspendLayout();
        //
        // label1
        //
        this.label1.AutoSize = true;
        this.label1.Font = new System.Drawing.Font("Microsoft Sans Serif", 24F,
        ((System.Drawing.FontStyle)((System.Drawing.FontStyle.Bold |
        System.Drawing.FontStyle.Italic))), System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        this.label1.Location = new System.Drawing.Point(148, 59);
        this.label1.Name = "label1";
        this.label1.Size = new System.Drawing.Size(561, 37);
        this.label1.TabIndex = 0;
        this.label1.Text = "Тренажер з теми \"Метод резолюції\"";
        //
        // label2
        //
        this.label2.AutoSize = true;
        this.label2.Font = new System.Drawing.Font("Microsoft Sans Serif", 22F,
        System.Drawing.FontStyle.Italic, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        this.label2.Location = new System.Drawing.Point(27, 118);
        this.label2.Name = "label2";
        this.label2.Size = new System.Drawing.Size(820, 36);
        this.label2.TabIndex = 1;
        this.label2.Text = "дистанційного навчального курсу \"Математична логіка\"";
        //
        // button1
        //
        this.button1.Font = new System.Drawing.Font("Microsoft Sans Serif", 14F,
        System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
        this.button1.Location = new System.Drawing.Point(117, 344);
    }
    #endregion
}

```

```

this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(635, 49);
this.button1.TabIndex = 2;
this.button1.Text = "Інформаційна сторінка";
this.button1.UseVisualStyleBackColor = true;
this.button1.Click += new System.EventHandler(this.button1_Click_1);
//
// button2
//
this.button2.Font = new System.Drawing.Font("Microsoft Sans Serif", 14F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.button2.Location = new System.Drawing.Point(117, 399);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(635, 53);
this.button2.TabIndex = 3;
this.button2.Text = "Перейти до тестування";
this.button2.UseVisualStyleBackColor = true;
this.button2.Click += new System.EventHandler(this.button2_Click);
//
// label3
//
this.label3.AutoSize = true;
this.label3.Font = new System.Drawing.Font("Microsoft Sans Serif", 14F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.label3.Location = new System.Drawing.Point(113, 217);
this.label3.Name = "label3";
this.label3.Size = new System.Drawing.Size(223, 24);
this.label3.TabIndex = 4;
this.label3.Text = "Розробив: Алексов С.В.";
//
// label4
//
this.label4.AutoSize = true;
this.label4.Font = new System.Drawing.Font("Microsoft Sans Serif", 14F,
System.Drawing.FontStyle.Regular, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.label4.Location = new System.Drawing.Point(113, 254);
this.label4.Name = "label4";
this.label4.Size = new System.Drawing.Size(496, 24);
this.label4.TabIndex = 5;
this.label4.Text = "Науковий керівник: К.Ф.-М.Н., доцент Парфьонкова Т.О.";

```

```

//
// Form1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(859, 475);
this.Controls.Add(this.label4);
this.Controls.Add(this.label3);
this.Controls.Add(this.button2);
this.Controls.Add(this.button1);
this.Controls.Add(this.label2);
this.Controls.Add(this.label1);
this.Name = "Form1";
this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
this.Text = "Тренажер";
this.ResumeLayout(false);
this.PerformLayout();

}

#endregion

private System.Windows.Forms.Label label1;
private System.Windows.Forms.Label label2;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Label label3;

private System.Windows.Forms.Label label4; }}

```

Додаток Б.

Вікно вибору теми

```

namespace TestRobAleksov
{
    partial class PractT1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise,
        false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify

```

```

/// the contents of this method with the code editor.
/// </summary>
private void InitializeComponent()
{
    this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
    this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
    this.ClientSize = new System.Drawing.Size(978, 558);
    this.Controls.Add(this.button3);
    this.Controls.Add(this.button2);
    this.Controls.Add(this.button1);
    this.Controls.Add(this.richTextBox4);
    this.Controls.Add(this.richTextBox3);
    this.Controls.Add(this.richTextBox2);
    this.Controls.Add(this.richTextBox1);
    this.Name = "PractT1";
    this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
    this.Text = "Тестування";
    this.ResumeLayout(false);

}

#endregion

private System.Windows.Forms.RichTextBox richTextBox1;
private System.Windows.Forms.RichTextBox richTextBox2;
private System.Windows.Forms.RichTextBox richTextBox3;
private System.Windows.Forms.RichTextBox richTextBox4;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Button button3;
}
}

```

Додаток В.

Виведення невірної довідки

```
private System.Windows.Forms.Button button1;  
private System.Windows.Forms.RichTextBox richTextBox1;  
private System.Windows.Forms.Label label1;  
private System.Windows.Forms.Label label2;  
}  
}
```

Додаток Г.

Вікно з кінцевим результатом та варіантами подальшого вибору

```
namespace TestRobAleksov
{
    partial class Pract1
    {
        /// <summary>
        /// Required designer variable.
        /// </summary>
        private System.ComponentModel.IContainer components = null;

        /// <summary>
        /// Clean up any resources being used.
        /// </summary>
        /// <param name="disposing">true if managed resources should be disposed; otherwise,
        false.</param>
        protected override void Dispose(bool disposing)
        {
            if (disposing && (components != null))
            {
                components.Dispose();
            }
            base.Dispose(disposing);
        }

        #region Windows Form Designer generated code

        /// <summary>
        /// Required method for Designer support - do not modify
        /// the contents of this method with the code editor.
        /// </summary>
        private void InitializeComponent()
        {
            this.richTextBox1 = new System.Windows.Forms.RichTextBox();
            this.richTextBox2 = new System.Windows.Forms.RichTextBox();
            this.richTextBox3 = new System.Windows.Forms.RichTextBox();
            this.richTextBox4 = new System.Windows.Forms.RichTextBox();
            this.button1 = new System.Windows.Forms.Button();
            this.button2 = new System.Windows.Forms.Button();
            this.button3 = new System.Windows.Forms.Button();
            this.SuspendLayout();
            //
            // richTextBox1
```

```

this.richTextBox1.Location = new System.Drawing.Point(12, 12);
this.richTextBox1.Name = "richTextBox1";
this.richTextBox1.Size = new System.Drawing.Size(954, 316);
this.richTextBox1.TabIndex = 0;
this.richTextBox1.Text = "";
//
// richTextBox2
// |
this.richTextBox2.Location = new System.Drawing.Point(12, 334);
this.richTextBox2.Name = "richTextBox2";
this.richTextBox2.Size = new System.Drawing.Size(304, 169);
this.richTextBox2.TabIndex = 1;
this.richTextBox2.Text = "";
//
// richTextBox3
//
this.richTextBox3.Location = new System.Drawing.Point(335, 334);
this.richTextBox3.Name = "richTextBox3";
this.richTextBox3.Size = new System.Drawing.Size(309, 169);
this.richTextBox3.TabIndex = 2;
this.richTextBox3.Text = "";
//
// richTextBox4
//
this.richTextBox4.Location = new System.Drawing.Point(664, 334);
this.richTextBox4.Name = "richTextBox4";
this.richTextBox4.Size = new System.Drawing.Size(302, 169);
this.richTextBox4.TabIndex = 3;
this.richTextBox4.Text = "";
//
// button1
//
this.button1.Font = new System.Drawing.Font("Microsoft Sans Serif", 10F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.button1.Location = new System.Drawing.Point(99, 509);
this.button1.Name = "button1";
this.button1.Size = new System.Drawing.Size(102, 37);
this.button1.TabIndex = 4;
this.button1.Text = "a");
this.button1.UseVisualStyleBackColor = true;
//
// button2
// |
this.button2.Font = new System.Drawing.Font("Microsoft Sans Serif", 10F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.button2.Location = new System.Drawing.Point(436, 509);
this.button2.Name = "button2";
this.button2.Size = new System.Drawing.Size(104, 37);
this.button2.TabIndex = 5;
this.button2.Text = "6");
this.button2.UseVisualStyleBackColor = true;
//
// button3
//
this.button3.Font = new System.Drawing.Font("Microsoft Sans Serif", 10F,
System.Drawing.FontStyle.Bold, System.Drawing.GraphicsUnit.Point, ((byte)(204)));
this.button3.Location = new System.Drawing.Point(765, 509);
this.button3.Name = "button3";
this.button3.Size = new System.Drawing.Size(97, 37);
this.button3.TabIndex = 6;
this.button3.Text = "B");
this.button3.UseVisualStyleBackColor = true;

```

```

//
// PractT1
//
this.AutoScaleDimensions = new System.Drawing.SizeF(6F, 13F);
this.AutoScaleMode = System.Windows.Forms.AutoScaleMode.Font;
this.ClientSize = new System.Drawing.Size(978, 558);
this.Controls.Add(this.button3);
this.Controls.Add(this.button2);
this.Controls.Add(this.button1);
this.Controls.Add(this.richTextBox4);
this.Controls.Add(this.richTextBox3);
this.Controls.Add(this.richTextBox2);
this.Controls.Add(this.richTextBox1);
this.Name = "PractT1";
this.StartPosition = System.Windows.Forms.FormStartPosition.CenterScreen;
this.Text = "Тестирование";
this.ResumeLayout(false);
}

#endregion

private System.Windows.Forms.RichTextBox richTextBox1;
private System.Windows.Forms.RichTextBox richTextBox2;
private System.Windows.Forms.RichTextBox richTextBox3;
private System.Windows.Forms.RichTextBox richTextBox4;
private System.Windows.Forms.Button button1;
private System.Windows.Forms.Button button2;
private System.Windows.Forms.Button button3;
}
}

```