

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

**РОЗРОБКА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ НАВЧАЛЬНИХ
ІНСТРУМЕНТІВ ПОБУДОВИ МАТЕМАТИЧНОЇ МОДЕЛІ
ОПТИМАЛЬНОГО РОЗКРОЮ МАТЕРІАЛІВ ДИСТАНЦІЙНОГО
КУРСУ «МЕТОДИ ОПТИМІЗАЦІЇ ТА ДОСЛІДЖЕННЯ
ОПЕРАЦІЙ»**

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

Виконавець роботи Серіков Ігор Ярославович
_____ «_____» _____ 202_ р.
(підпис)

Науковий керівник Парфьонова Тетяна Романівна
_____ «_____» _____ 202_ р.
(підпис)

Рецензент _____

ПОЛТАВА 2024

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ.....	3
ВСТУП.....	4
РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ	6
РОЗДІЛ 2. ІНФОРМАЦІЙНИЙ ОГЛЯД.....	7
2.1. Огляд існуючих програм.....	7
3. ТЕОРЕТИЧНА ЧАСТИНА.....	10
3.1. Методологія розробки.....	10
3.2 . Алгоритм роботи.....	16
3.3 Блок-схема системи тестування знань	21
4. ПРАКТИЧНА ЧАСТИНА.....	23
4.1 .Обґрунтування вибору програмних засобів	23
4.2. Опис програмної реалізації.....	24
4.3. Опис роботи тренажеру	29
ВИСНОВКИ	37
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ	39
ДОДАТОК А. КОД ПРОГРАМИ	40

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ, СКОРОЧЕНЬ, ТЕРМІНІВ

Умовні позначення, символи, скорочення, терміни	Пояснення умовних позначень, скорочень, символів
Pycharm	PyCharm – це інтегроване середовище розробки (IDE) для мови програмування Python, розроблене компанією JetBrains. Воно надає розробникам потужні інструменти та функції для написання, тестування та налагодження коду на Python. PyCharm є одним з найпопулярніших IDE для Python завдяки своїй функціональності та зручності у використанні.
Message	У мові програмування Python використовується для текстового повідомлення, яке відображає багаторядковий текст
Frame	У мові програмування, такій як Python (Py), використовується для створення вікна програми та графічних елементів
Lambda	Це метод, який зазвичай використовується для анонімних функцій для передачі параметрів у функції обробки подій

ВСТУП

Методи оптимізації та дослідження операцій (ОП) відіграють ключову роль у сучасних наукових дослідженнях і практичному застосуванні в різних галузях. Вони пропонують систематичний підхід до вирішення складних задач прийняття рішень, які виникають в інженерії, економіці, управлінні, логістиці, фінансах та багатьох інших сферах. Основна мета методів оптимізації полягає в знаходженні найкращих можливих рішень, що максимізують або мінімізують певні цільові функції при дотриманні заданих обмежень.

Дослідження операцій, з іншого боку, використовують математичні моделі, статистичний аналіз та алгоритмічні методи для аналізу та оптимізації складних систем. Це дозволяє організаціям та підприємствам підвищити ефективність, зменшити витрати та покращити якість послуг або продукції. ОП охоплює широкий спектр методів, включаючи лінійне та нелінійне програмування, динамічне програмування, теорію ігор, теорію черг, евристичні алгоритми та багато інших.

Ці методи стали невід'ємною частиною сучасного менеджменту та стратегічного планування, забезпечуючи інструменти для аналізу, моделювання та вирішення проблем, які раніше здавалися непосильними. Завдяки постійному розвитку комп'ютерних технологій та алгоритмів, методи оптимізації та дослідження операцій продовжують вдосконалюватися, відкриваючи нові можливості для ефективного управління та прийняття рішень у різних галузях.

Об'єкт розробки – алгоритм, блок-схема, розробка програмного забезпечення системи тестування знань.

Предмет розробки – навчальний інструмент дистанційного навчального курсу «Методи оптимізації та дослідження операцій» для

вивчення побудови математичної моделі оптимального розкрою матеріалів та розробка його програмного забезпечення.

Кваліфікаційна робота складається з чотирьох розділів. У першому розділі описано постановку задачі для реалізації тренажеру. У другому розділі розглянуто існуючі комп'ютерні рішення. У третьому розділі описано методологію розробки, детально описаний алгоритм роботи та блок-схема роботи системи тестування знань. У четвертому розділі описано процес програмної реалізації тренажеру та опис роботи алгоритму. Обсяг пояснювальної записки: 44 стор., в т.ч. основна частина – 36 стор., джерела – 9 назв.

РОЗДІЛ 1. ПОСТАНОВКА ЗАДАЧІ

Основною метою кваліфікаційної роботи є розробка програмного забезпечення для впровадження тестування знань з побудови математичної моделі оптимального розкрою матеріалів для дистанційного курсу "Методи оптимізації та дослідження операцій". Основні завдання включають наступне:

1. Освоєння матеріалу за темою «Побудова математичних моделей» курсу «Методи оптимізації та дослідження операцій».
2. Розробка ефективного алгоритму для реалізації системи тестування знань.
3. Створення блок-схеми для наочного представлення алгоритму системи тестування знань.
4. Реалізація програмних компонентів, необхідних для операційної роботи тренажеру.
5. Докладний опис отриманих результатів під час розробки програмних елементів системи тестування знань.

У процесі створення програмного забезпечення для тренажеру знань важливо враховувати такі ключові аспекти:

1. Стартове вікно системи тестування знань, яке містить інформацію про тематику тестування, кнопку для початку тесту та інформацію про розробника та наукового керівника.
2. Інші вікна системи, що включають тестові завдання та питання з теми тестування, на які користувач має надавати вірні або невірні відповіді.

РОЗДІЛ 2. ІНФОРМАЦІЙНИЙ ОГЛЯД

2.1. Огляд існуючих програм

Програми для розв'язку задач з методів оптимізації та дослідження операцій є спеціалізованими інструментами, які використовуються для вирішення широкого спектра задач в інженерії, економіці, управлінні, логістиці та інших галузях.

Розглянемо детальніше інструмент для розв'язку задач.

MATLAB забезпечує широкі можливості для виконання різноманітних математичних обчислень, включаючи лінійну алгебру, статистику, обробку сигналів, оптимізацію та інші математичні операції. Вбудовані функції дозволяють швидко і точно виконувати складні обчислення, що є критично важливим для наукових досліджень та інженерії.

MATLAB дозволяє створювати та аналізувати математичні моделі реальних систем. Інструмент Simulink, який інтегрований з MATLAB, забезпечує можливість моделювання, симуляції та аналізу динамічних систем, що робить його незамінним інструментом для інженерів та науковців.

MATLAB надає потужні засоби для аналізу та обробки даних. Інструменти для статистичного аналізу, візуалізації даних та машинного навчання дозволяють дослідникам швидко отримувати інсайти з великих обсягів даних.

MATLAB має багаті можливості для візуалізації даних. Графічні інструменти дозволяють створювати дво- та тривимірні графіки, візуалізувати результати симуляцій та створювати інтерактивні графічні інтерфейси.

MATLAB підтримує як імперативне, так і об'єктно-орієнтоване програмування. Мова MATLAB є простою для вивчення і використання, що дозволяє дослідникам та інженерам швидко створювати скрипти та функції для автоматизації своїх задач.

Застосування MATLAB

MATLAB широко використовується в різних галузях інженерії для проектування, аналізу та оптимізації систем. Наприклад, у електроніці MATLAB застосовується для обробки сигналів та аналізу систем зв'язку, а в механіці – для моделювання динаміки та управління механічними системами.

У фінансовій сфері MATLAB використовується для моделювання фінансових ринків, аналізу ризиків, оптимізації інвестиційного портфеля та розробки алгоритмів для алгоритмічної торгівлі.

Науковці у різних галузях використовують MATLAB для обробки та аналізу експериментальних даних, моделювання природних явищ та розробки нових методів та алгоритмів.

MATLAB є потужним інструментом для навчання та досліджень у вищих навчальних закладах. В університетах MATLAB використовується для викладання математики, фізики, інженерії та комп'ютерних наук.

Переваги Використання MATLAB

1.Швидкий Розвиток Прототипів. Завдяки інтерактивному середовищу MATLAB дослідники можуть швидко розробляти, тестувати та вдосконалювати свої ідеї. Це дозволяє значно скоротити час від початкової концепції до робочого прототипу.

2.Багата Бібліотека Функцій. MATLAB надає доступ до величезної кількості вбудованих функцій та інструментів, що дозволяє дослідникам зосередитися на своїх завданнях, а не на розробці базових алгоритмів.

3.Інтеграція з Іншими Мовами Програмування. MATLAB легко інтегрується з іншими мовами програмування, такими як C/C++, Python, Java та .NET, що дозволяє використовувати його разом з іншими інструментами та бібліотеками.

4. Підтримка та Спільнота. MATLAB має велику спільноту користувачів та розробників, а також обширну документацію та навчальні матеріали, що полегшує процес вивчення та використання цієї мови.

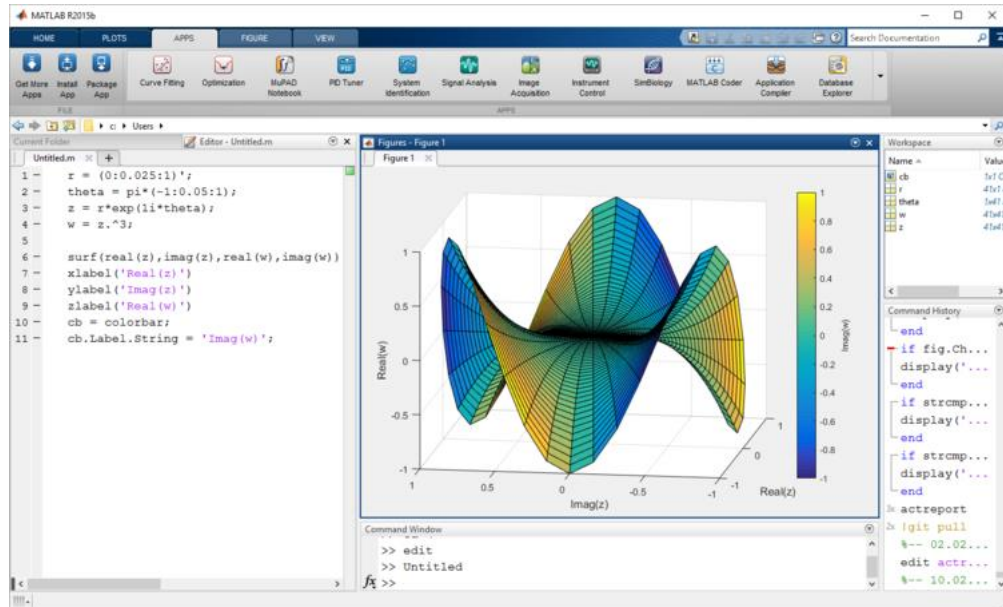


Рисунок 2.1 – Середовище MATLAB.

3. ТЕОРЕТИЧНА ЧАСТИНА

3.1. Методологія розробки

Розробка якісного продукту починається з визначення його життєвого циклу, що є чітким планом дій. Це дозволяє розробникам зрозуміти кінцевий результат, методи його досягнення та інструменти, які необхідно використовувати. Методологія розробки програмного забезпечення охоплює перевірені способи та практики, що дозволяють створити діджитал-продукт правильно та якісно.

Серед найвідоміших методологій розроблення програмного забезпечення можна виділити Rational Unified Process (RUP). Ця методологія пропонує ітеративну модель розроблення, яка складається з чотирьох фаз: початок, дослідження, побудова та впровадження. Кожен цикл розроблення завершується генерацією працездатної версії системи. У ході супроводу продукт продовжує розвиватися і знову проходить ті самі фази. Розроблення програмного забезпечення на базі RUP базується на створенні та супроводі моделей на основі UML.

Інша методологія, Microsoft Solution Framework (MSF), була розроблена для підвищення керованості процесів розроблення всередині корпорації Microsoft, але пізніше стала використовуватися розробниками, які застосовують продукти Microsoft. MSF включає чотири фази: аналіз, проектування, розробка, стабілізація. Вона є ітераційною і припускає використання об'єктно-орієнтованого моделювання. На відміну від RUP, MSF більше орієнтована на розроблення бізнес-додатків.

Методологія eXtreme Programming (XP) надає основну увагу ефективній комунікації між замовником і виконавцем протягом усього проекту. Основу підходу складає командна робота та постійне тестування прототипів, що дозволяє швидко відслідковувати зміни вимог.

Ці методології допомагають структурувати процес розробки, забезпечуючи систематичний підхід до створення програмного забезпечення, який дозволяє враховувати потреби замовника та забезпечувати високу якість кінцевого продукту.

Гнучке розроблення програмного забезпечення (Agile) – це клас методологій, які базуються на ітеративній моделі життєвого циклу. У цьому підході вимоги та рішення еволюціонують завдяки співпраці між самоорганізовуваними командами.

Об'єктно-орієнтована методологія розробки (OOMP) є підходом до програмування, що ґрунтується на концепції об'єктів і класів. Вона дозволяє розглядати програмне забезпечення як колекцію об'єктів, які взаємодіють між собою для виконання певних завдань. OOMP спрямована на полегшення розробки, управління та розуміння складних систем шляхом використання об'єктно-орієнтованих концепцій.

Основні принципи об'єктно-орієнтованої методології розробки включають інкапсуляцію, яка полягає в тому, що об'єкти мають внутрішній стан і поведінку, які вони приховують від зовнішнього світу. Всі зміни стану об'єкта мають відбуватися за допомогою відповідних методів, що контролюють доступ до даних. Наслідування дозволяє об'єктам успадковувати властивості та методи від інших об'єктів, спрощуючи повторне використання коду та забезпечуючи ієрархічну структуру класів. Поліморфізм означає, що об'єкти можуть виконувати однакові дії, але різними способами. Наприклад, одна й та сама операція може мати різні реалізації залежно від типу об'єкта. Абстракція дозволяє моделювати реальний світ у вигляді об'єктів з властивостями та методами, які відображають важливі аспекти предметної області, і приховувати від програміста непотрібні деталі реалізації.

OOMP також передбачає використання класів, об'єктів, спадкування, поліморфізму, інтерфейсів та інших конструкцій для створення структурованих, легко змінюваних і розширюваних програмних систем.

Кожен клас може мати свої властивості (дані) та методи (функції), які визначають його поведінку. Взаємодія між об'єктами відбувається через виклик методів одного об'єкта з іншого. ООМР дозволяє розробникам легко створювати програми, підвищуючи їхню якість і обслуговуваність, зменшуючи залежність між різними частинами коду та сприяючи повторному використанню коду.

Процедурне програмування полягає у розбитті великої програми на сукупність процедур або підпрограм. Одна з підпрограм повинна бути головною, і саме з неї починається виконання програми. Процедурний підхід потребує структуризації майбутньої програми, розділення її на окремі процедури. При розробці окремої процедури про інші процедури потрібно знати тільки їхнє призначення та спосіб виклику.

Структурне програмування — це підхід до програмування, при якому програма в цілому і окремі процедури розглядаються як послідовності канонічних структур: лінійних ділянок, циклів і розгалужень.

Дані представляють собою факти та ідеї у формалізованому вигляді, придатному для передачі і обробки у процесі. Інформація — це сенс, який додається даним при їх поданні.

Обробка даних полягає у виконанні систематичної послідовності дій з даними. Дані зберігаються на носіях даних, а сукупність носіїв, що використовуються при обробці даних, називається інформаційним середовищем. Набір даних, що містяться в інформаційному середовищі в будь-який момент, називається станом цього середовища. Процес можна визначити як послідовність станів інформаційного середовища, що змінюють один одного.

Описати процес означає визначити послідовність станів заданого інформаційного середовища. Щоб процес автоматично виконувався на ПК, його опис має бути формалізованим, тобто програмою. Програми створюються з розрахунку на їхнє використання людьми, які не брали участі в розробці (користувачами). Для освоєння програми користувачем, окрім її

тексту, потрібна додаткова документація. Програма або логічно зв'язана сукупність програм на носіях даних, забезпечена документацією, називається програмним забезпеченням (ПЗ). Програма дозволяє автоматично обробляти дані на комп'ютері, а документація допомагає зрозуміти, які функції виконує програма, як підготувати початкові дані і запустити програму, а також що означають отримані результати. Документація також корисна при модифікації програми.

Продуктом технології програмування є ПЗ, що містить програми, які виконують необхідні функції. Під програмою часто розуміють правильну програму, тобто таку, що не містить помилок. Надійність ПЗ визначається його здатністю безвідмовно виконувати певні функції за заданих умов протягом певного періоду часу з великою ймовірністю. Відмова в ПЗ означає прояв у ньому помилки.

Технологія програмування це сукупність виробничих процесів, що приводить до створення необхідного ПЗ, а також опис цієї сукупності процесів.

Не слід також плутати технологію програмування з методологією програмування. В обох випадках вивчаються методи, але в технології програмування методи розглядаються "зверху" - з погляду організації технологічних процесів, а в методології програмування методи розглядаються "знизу" - з погляду основ їх побудови

- Неформальний характер вимог до ПЗ (постановки завдання), але формалізований основний об'єкт розробки - програми ПЗ.
- Творчий характер розробки ПЗ (на кожному кроці доводиться робити який-небудь вибір, ухвалювати яке-небудь рішення).
- Програмний продукт є деякою сукупністю текстів (тобто статичних об'єктів), сенс же (семантика) цих текстів виражається процесами обробки даних і діями користувачів, що запускають ці процеси (тобто є динамічним).

- Продукт розробки має і іншу специфічну особливість: ПЗ при своєму використанні (експлуатації) не витрачається і не витрачає використовуваних ресурсів.

Під життєвим циклом програмного забезпечення (ПЗ) розуміють весь період його розробки та експлуатації, починаючи від задуму і закінчуючи припиненням використання. Життєвий цикл включає всі процеси створення і використання ПЗ.

Розрізняють такі стадії життєвого циклу ПЗ: розробка, виробництво програмних виробів (ПВ) і експлуатація. Стадія розробки складається з етапу зовнішнього опису, конструювання, кодування і атестації.

Зовнішній опис ПЗ визначає його поведінку і фіксує вимоги щодо якості. Конструювання включає розробку архітектури, структури програм і їхню детальну специфікацію. Кодування означає створення текстів програм на мовах програмування та їхню відладку з тестуванням. Атестація ПЗ полягає в оцінці його якості, після чого розробка вважається завершеною.

Програмний виріб (ПВ) — це екземпляр або копія ПЗ. Стадія експлуатації охоплює процеси зберігання, впровадження, супроводу, транспортування і застосування ПЗ. Вона складається з фази застосування і фази супроводу. Застосування означає використання ПЗ для вирішення практичних завдань, а супровід передбачає збір інформації про якість ПЗ, усунення помилок, модифікацію та інформування користувачів про зміни.

Якість ПЗ визначається його здатністю задовольняти потреби користувачів. Основні критерії якості включають: функціональність, надійність, легкість застосування, ефективність, супровід і мобільність.

Функціональність — це здатність ПЗ виконувати необхідні функції, що визначені у зовнішньому описі. Надійність означає здатність ПЗ безвідмовно виконувати функції за певних умов протягом визначеного часу. Легкість застосування передбачає мінімізацію зусиль користувача щодо підготовки даних і використання ПЗ. Ефективність — це співвідношення

рівня послуг, що надаються ПЗ, до використаних ресурсів. Супровід означає можливість легко вносити зміни для усунення помилок і адаптації ПЗ до змінюваних потреб. Мобільність — здатність ПЗ працювати в різних середовищах.

Забезпечення надійності ПЗ передбачає попередження помилок, самовиявлення та самовиправлення помилок, забезпечення стійкості до помилок. Це включає боротьбу зі складністю, точність перекладу, подолання бар'єру між користувачем і розробником та контроль ухвалюваних рішень.

Для досягнення високої якості ПЗ важливо дотримуватись загальних принципів забезпечення надійності, таких як:

- попередження помилок;
- самовиявлення і самовиправлення помилок;
- забезпечення стійкості до помилок;
- забезпечення незалежності компонент системи;
- використання ієрархічних структур.

Попередження помилок спрямоване на боротьбу зі складністю та забезпечення точності перекладу вимог. Самовиявлення і самовиправлення помилок означає, що програма повинна містити засоби для виявлення та виправлення помилок під час виконання. Стійкість до помилок передбачає локалізацію області впливу помилок і зменшення їх негативних наслідків.

Забезпечення незалежності компонент системи означає мінімізацію зв'язків між частинами системи, що досягається модульним програмуванням. Використання ієрархічних структур допомагає локалізувати зв'язки між компонентами на різних рівнях ієрархії, що дозволяє розбивати велику систему на підсистеми для спрощення розробки і управління.

Таким чином, розробка ПЗ передбачає дотримання ряду принципів і етапів, що забезпечують його функціональність, надійність і ефективність в експлуатації.

3.2 . Алгоритм роботи

Загальний алгоритм роботи додатку:

Крок 1: Запуск системи;

Крок 2: Вивід стартового екрану;

Крок 3: По бажанню перегляд інформації про алгоритм;

Крок 4: Ознайомлення користувача із завданням;

Крок 5: Проходження користувачем завдань у вигляді тестів

Крок 6: Вивід вікна тема завершена та результати тестування;

Крок 7: Завершення додатку.

Алгоритм роботи коду додатку:

В момент запуску програми, на екрані з'являється стартове вікно, після натиснення користувачем кнопки “Перейти до завдання”, користувач отримує умови завдання: При виготовленні парників, використовується матеріал у вигляді металевих стрижнів довжиною 220 см. Цей матеріал розрізається на стрижні довжиною 120, 100 і 70 см. Для виконання замовлення потрібно виготовити 80 стрижнів довжиною 120 сантиметрів , 120 стрижнів довжиною 100 сантиметрів і 102 стрижня довжиною 70 см.

Визначити:

1. Скільки існує раціональних способів розкрою
2. Побудувати математичну модель задачі мінімізації кількості матеріалу, який необхідно розрізати, щоб виконати замовлення

Крок 1. На екрані запитання, “Виберіть правильні варіанти розкрою, які можна використовувати згідно умови задачі ” також запропоновані варіанти відповіді:

- а) 1 деталь 120 см та 1 деталь 100 см

- b) 2 деталі 100 см та 1 деталь 70 см
- c) 1 деталь 100 см та 1 деталь 70 см
- d) 1 деталь 100 см та 2 деталі 70 см
- e) 2 деталі 100 см
- f) 1 деталь 120 см та 1 деталь 70 см
- g) всі по одній
- h) 3 деталі 70 см
- i) 2 деталі 70 см і дві 100 см
- j) дві деталі 70 , 1 деталь 100 см і 1 деталь 120 см

Користувач вибирає декілька правильних варіантів відповідей, після чого користувач переходить на другий крок якщо всі відповіді правильні. Інакше з'являється текстове повідомлення помилкового вибору “Сума довжин деталей не може перевищувати 220 см”. Тож правильний набір відповідей : А, В, Д, Е, Ж. Таким чином користувач отримує кількість способів оптимального розкрою - “5”

Крок 2. На екрані з'являється наступне питання “ Побудуємо математичну модель даної задачі. Яке із позначень змінної, варто використовувати в цій задачі”. Нижче варіанти відповіді:

- a) X_{ij}
- b) X_i
- c) X

Правильна відповідь: b

Якщо вибір інший, то на екрані з'являється повідомлення: “Вибір помилковий X_{ij} можна використовувати для позначення кількості одиниць матеріалу, i - десятину для розрізу j – м способом. Так як у даній задачі матеріал одного типу, то можна використовувати позначення X_i – кількість одиниць матеріалу для розрізу i - м способом “

Перехід на крок 3. Враховуючи що X_i – кількість одиниць матеріалу для розрізу i – м способом, вибрати вираз що відповідає цільовій функції.

- a) $X_1 + 2X_3 + X_4$

b) $80X_1 + 120X_3 + 120X_4 + X_5 + X_6$

c) $X_1 + X_2 + X_3 + X_4 + X_5$

Правильна відповідь – с

Якщо помилка то на екрані з'явиться повідомлення : “Вибір помилковий. Так як згідно умови необхідно оптимізувати кількість матеріалу, що необхідно розрізати для виконання замовлення, то цільова функція – це сума кількостей одиниць матеріалу для розрізу і – м способом.”

Перехід до кроку 4.

Крок 4. На екрані з'являється питання: “Цільова функція даної задачі”.

Варіанти відповіді:

a) Максимізується

b) Мінімізується

c) Дорівнює 220

Правильна відповідь – b

Якщо вибір інший , то з'явиться повідомлення про помилку: “Вибір помилковий. Згідно умови необхідно мінімізувати кількість матеріалу який потрібно розрізати для виконання замовлення.”

Перехід до кроку 5

Крок 5. На екрані з'являється питання: “Побудуємо систему обмежень. Обмеження на кількість стрижнів довжиною 120 см має вигляд:”

Варіанти відповіді:

a) $x_1 + x_4 \geq 80$

b) $x_1 + x_2 \leq 80$

c) $x_1 + x_2 + x_3 \geq 80$

d) $x_1 + 2x_3 + x_4 \geq 120$

e) $x_1 + x_2 + x_5 \geq 102$

Правильна відповідь – а

Якщо обрано інше, то повідомлення про помилку має вигляд: “Вибір помилковий так як деталі довжиною 120 см використовуються у першому та четвертому способах розкрою.”. Перехід до 6 кроку

Крок 6. На екрані з'являється питання: “Обмеження на кількість стрижнів довжиною 100 см має вигляд:”

Варіанти відповіді:

- a) $x_1 + x_4 \geq 80$
- b) $x_1 + x_2 \leq 80$
- c) $x_1 + x_2 + x_3 \geq 80$
- d) $x_1 + x_2 + 2x_3 \geq 120$
- e) $x_1 + x_2 + x_5 \geq 102$

Правильна відповідь – d

Якщо обрано інше, то повідомлення про помилку має вигляд: “Вибір помилковий, так як деталі довжиною 100 см використовуються у 1-му, 2-му та 3-му способах розкрою.” Перехід на 7 крок.

Крок 7. На екрані з'являється питання: “ Обмеження на кількість стрижнів довжиною 70 см має вигляд:”

Варіанти відповіді:

- a) $x_1 + x_4 \geq 80$
- b) $x_2 + x_4 + 3x_5 \leq 102$
- c) $x_1 + x_2 + x_3 \geq 80$
- d) $x_1 + x_2 + 2x_3 \geq 120$
- e) $x_1 + x_2 + x_5 \geq 102$

Правильна відповідь – b

Якщо вибрано не вірно то користувач отримає повідомлення : “Вибір помилковий, так як деталі довжиною 70 см використовуються у 2-му, 4-му, та 5 – му способах розкрою”. Перехід до кроку 8

Крок 8. На екрані з'являється питання: “Змінні x_i , $i = 1,2,3,4,5$ мають відповідати умові: “

Варіанти відповіді:

- a) $x_i \geq 0, i = \overline{1,5}$
- b) $x_i < 0, i = \overline{1,5}$

$$c) \ x_i \geq 0, i = \overline{1,5}$$

$$d) \ x_i > 0, i = \overline{1,5}$$

Правильна відповідь : - а

Якщо вибрано інше то користувач отримує повідомлення :“Кількість стрижнів не може бути від’ємною”

Далі з’являється текст: “Отже математична модель задачі наступна:

Цільова функція: $X_1 + X_2 + X_3 + X_4 + X_5 \rightarrow \min$

Система обмежень :

$$\begin{cases} x_1 + x_4 \geq 80, \\ x_1 + x_2 + 2x_3 \geq 120 \\ x_2 + x_4 + 3x_5 \geq 102 \\ x_i \geq 0, i = \overline{1,5} \end{cases}$$

Алгоритм завершено.

Після кожного запуску програми, питання для користувача подаються випадково.

3.3 Блок-схема системи тестування знань

Блок-схема системи (рис. 3.1)

4. ПРАКТИЧНА ЧАСТИНА

4.1. Обґрунтування вибору програмних засобів

Python — це дуже популярна мова програмування, яка має декілька переваг, що роблять її привабливим вибором для реалізації програм. Ось деякі з найбільш вагомих причин:

1. Простота вивчення: Python відомий своєю простотою та зрозумілістю. Синтаксис Python легкий для вивчення і зрозуміння, що робить його ідеальним вибором для початківців.
2. Широке співробітництво: Python має активну спільноту розробників, що означає наявність безлічі бібліотек та модулів для різних цілей. Це полегшує створення програм із складними функціями, не переписуючи все з нуля.
3. Кросплатформенність: Python підтримується на різних операційних системах, таких як Windows, macOS і Linux. Це означає, що ви можете розробляти програми на Python і запускати їх на будь-якій платформі.
4. Розширюваність: Python легко поєднується з іншими мовами програмування, такими як C, C++, а також з іншими технологіями. Ви можете використовувати Python для написання швидкодіючих компонентів програми та взаємодії з ними.
5. Велика кількість бібліотек: У Python є велика кількість бібліотек для різних завдань, таких як обробка даних, машинне навчання, веб-розробка і багато інших. Це дозволяє розробникам швидко створювати програми з різноманітними функціональними можливостями.

Загалом, Python — це потужний і універсальний інструмент для розробки програм, який дозволяє створювати широкий спектр застосувань, від простих скриптів до великих проектів.

4.2. Опис програмної реалізації

Програмні компоненти тренажеру програмно реалізовані на основі мови програмування Python.(рис. 4.1)

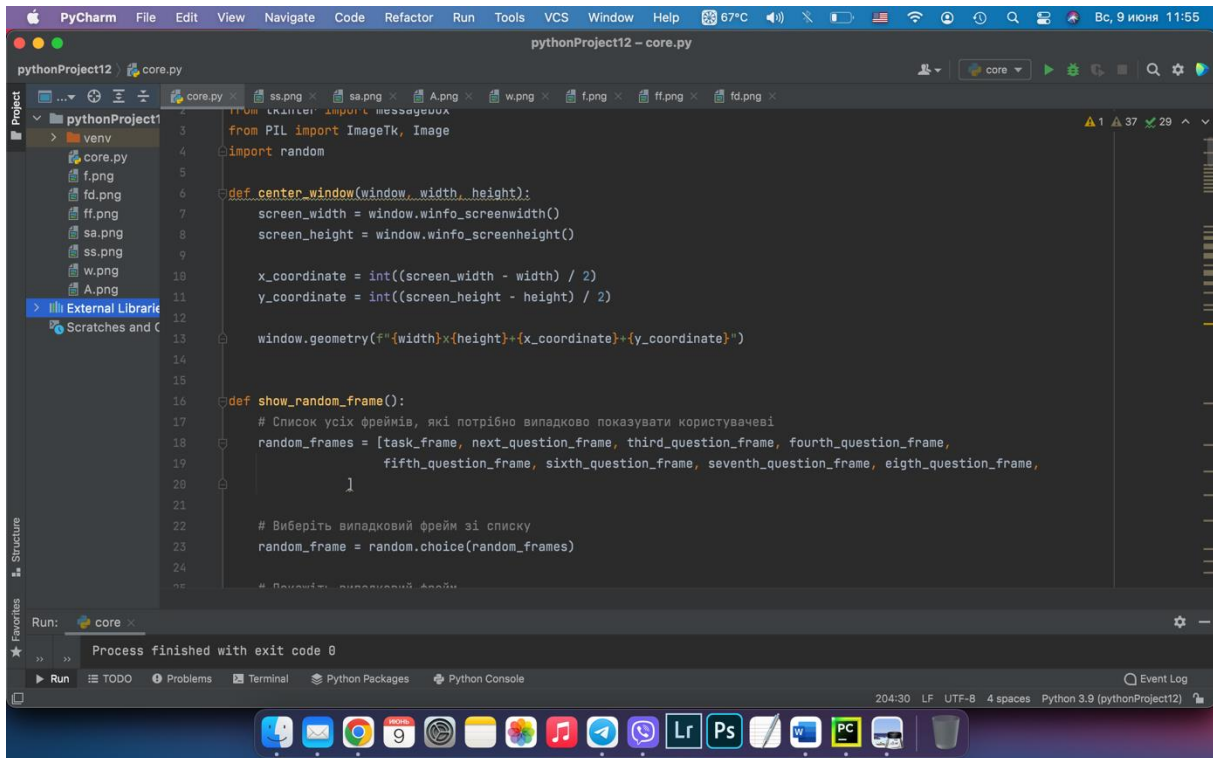


Рисунок 4.1 – Середовище розробки додатків PyCharm

Так як було вирішено створювати інтерфейс саме в середині програмного коду, то код початкового вікна виглядає так(рис. 4.2)



Рисунок 4.2 – Головний екран системи тестування знань.

Після відображення початкового екрану, наступним етапом є написання програмного коду для питання по задачі, та першого тестового питання. (рис. 4.3)

Задача розкрою матеріалу

Умова задачі:

При виготовленні парників, використовується матеріал у вигляді металевих стрижнів довжиною 220 см. Цей матеріал розрізається на стрижні довжиною 120, 100 і 70 см. Для виконання замовлення потрібно виготовити 80 стрижнів довжиною 120 сантиметрів, 120 стрижнів довжиною 100 сантиметрів і 102 стрижня довжиною 70 см.

Визначити:

1. Скільки існує раціональних способів розкрою
2. Побудувати математичну модель задачі мінімізації кількості матеріалу, який необхідно розрізати, щоб виконати замовлення

Виберіть правильні варіанти розкрою, які можна використовувати згідно умови задачі:

- ☐ 1 деталь 120 см та 1 деталь 100 см
- ☐ 2 деталі 100 см та 1 деталь 70 см
- ☐ 1 деталь 100 см та 1 деталь 70 см
- ☐ 1 деталь 100 см та 2 деталі 70 см
- ☐ 2 деталі 100 см
- ☐ 1 деталь 120 см та 1 деталь 70 см
- ☐ всі по одній
- ☐ 3 деталі 70 см
- ☐ 2 деталі 70 см і дві 100 см
- ☐ дві деталі 70 , 1 деталь 100 см і 1 деталь 120 см

Перевірити

Рисунок 4.3 – Перше питання з лекційного матеріалу.

Після першого вікна реалізується наступне випадкове вікно. (рис. 4.4)

Задача розкрою матеріалу

Обмеження на кількість стрижнів довжиною 70 см має вигляд:

a) $x_1 + x_4 \geq 80$

b) $x_2 + x_4 + 3x_5 \leq 102$

c) $x_1 + x_2 + x_3 \geq 80$

d) $x_1 + x_2 + 2x_3 \geq 120$

e) $x_1 + x_2 + x_5 \geq 102$

Перевірити

Рисунок 4.4 – Наступне вікно.

Весь вміст тренажеру запрограмовано за допомогою програмних засобів Python. (рис. 4.5)

```

from PIL import ImageTk, Image
import random

def center_window(window, width, height):
    screen_width = window.winfo_screenwidth()
    screen_height = window.winfo_screenheight()

    x_coordinate = int((screen_width - width) / 2)
    y_coordinate = int((screen_height - height) / 2)

    window.geometry(f"{width}x{height}+{x_coordinate}+{y_coordinate}")

def show_random_frame():
    # Список усіх фреймів, які потрібно випадково показувати користувачеві
    random_frames = [task_frame, next_question_frame, third_question_frame, fourth_question_frame,
                     fifth_question_frame, sixth_question_frame, seventh_question_frame, eighth_question_frame,
                     1]

    # Вибір випадкового фрейму зі списку
    random_frame = random.choice(random_frames)

    # Показати випадковий фрейм

```

Рисунок 4.5 – Код тренажеру.

Додатково реалізовано показ варіантів відповіді, через фото. (рис. 4.6)

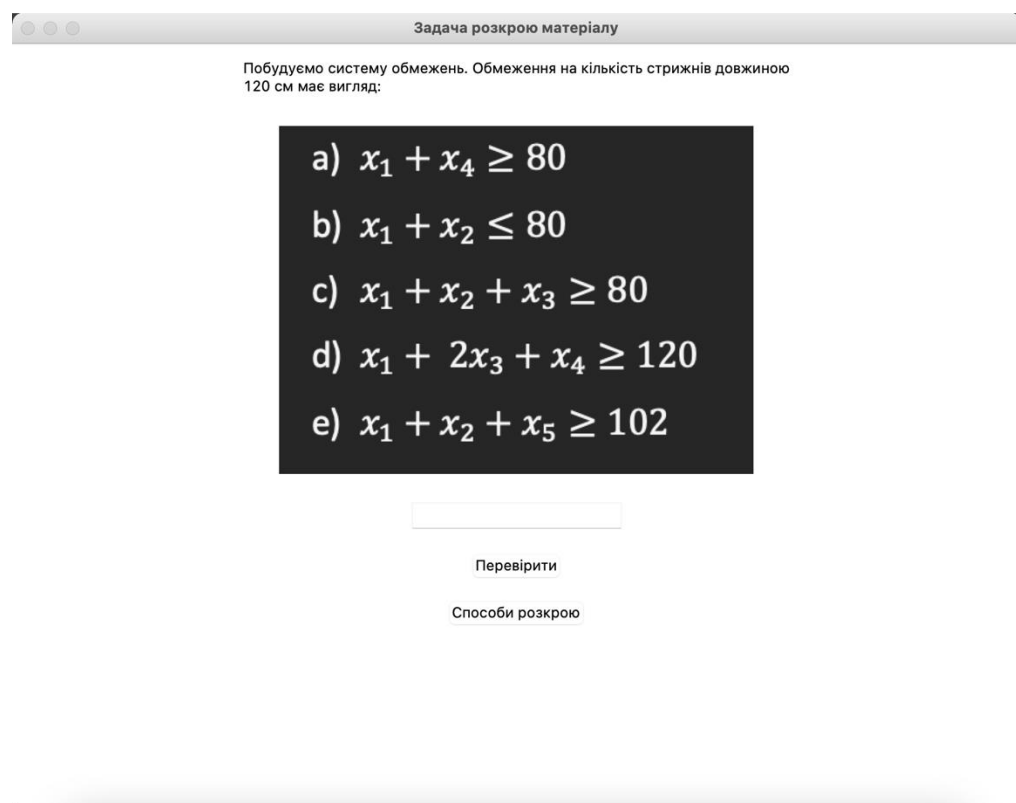


Рисунок 4.6 – Вікно інформації

4.3. Опис роботи тренажеру

Після запуску програми, користувача вітає головний екран. (рис. 4.7)



Рисунок 4.7 – Головний екран системи тестування знань.

Далі, користувач натискає кнопку перейти до завдання, та отримує перше тестове завдання з умовою задачі. (рис. 4.8)

Задача розкрою матеріалу

Умова задачі:

При виготовленні парників, використовується матеріал у вигляді металевих стрижнів довжиною 220 см. Цей матеріал розрізається на стрижні довжиною 120, 100 і 70 см. Для виконання замовлення потрібно виготовити 80 стрижнів довжиною 120 сантиметрів, 120 стрижнів довжиною 100 сантиметрів і 102 стрижня довжиною 70 см.

Визначити:

1. Скільки існує раціональних способів розкрою
2. Побудувати математичну модель задачі мінімізації кількості матеріалу, який необхідно розрізати, щоб виконати замовлення

Виберіть правильні варіанти розкрою, які можна використовувати згідно умови задачі:

- ☐ 1 деталь 120 см та 1 деталь 100 см
- ☐ 2 деталі 100 см та 1 деталь 70 см
- ☐ 1 деталь 100 см та 1 деталь 70 см
- ☐ 1 деталь 100 см та 2 деталі 70 см
- ☐ 2 деталі 100 см
- ☐ 1 деталь 120 см та 1 деталь 70 см
- ☐ всі по одній
- ☐ 3 деталі 70 см
- ☐ 2 деталі 70 см і дві 100 см
- ☐ дві деталі 70 , 1 деталь 100 см і 1 деталь 120 см

Перевірити

Рисунок 4.8 – Перше завдання.

Після того, як користувач обрав свій варіант відповіді, тренажер перевіряє цю відповідь і надає додаткову інформацію користувачеві через відповідне вікно. (рис. 4.9)

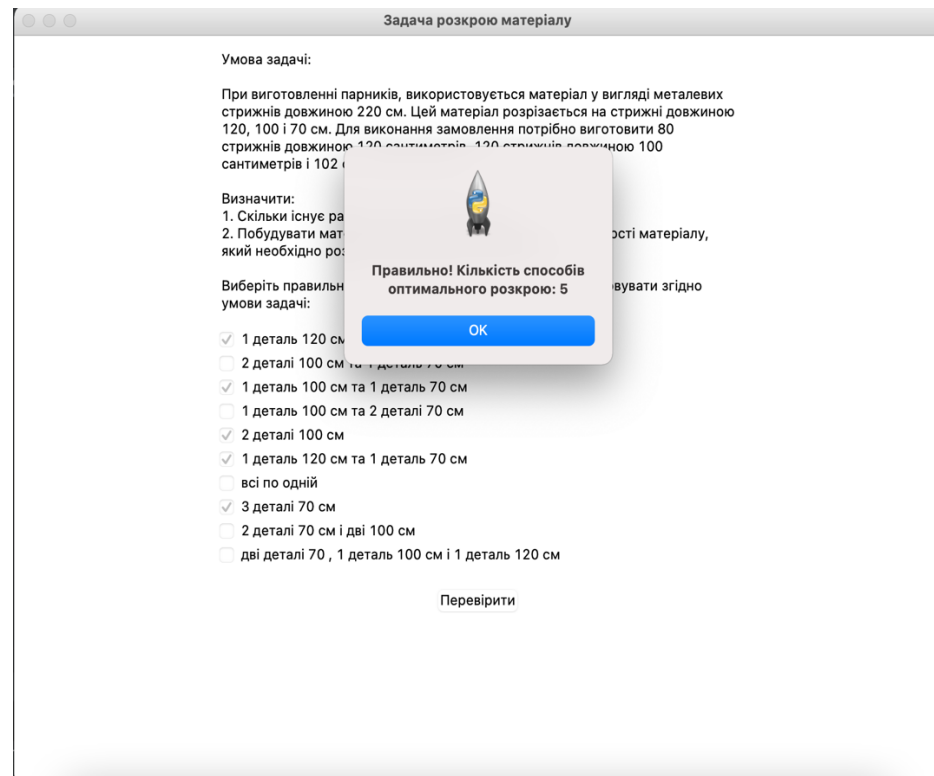


Рисунок 4.9 – Повідомлення у випадку правильної відповіді.

У разі неправильної відповіді користувача, тренажер відображає наступне попередження. (рис. 5.0)

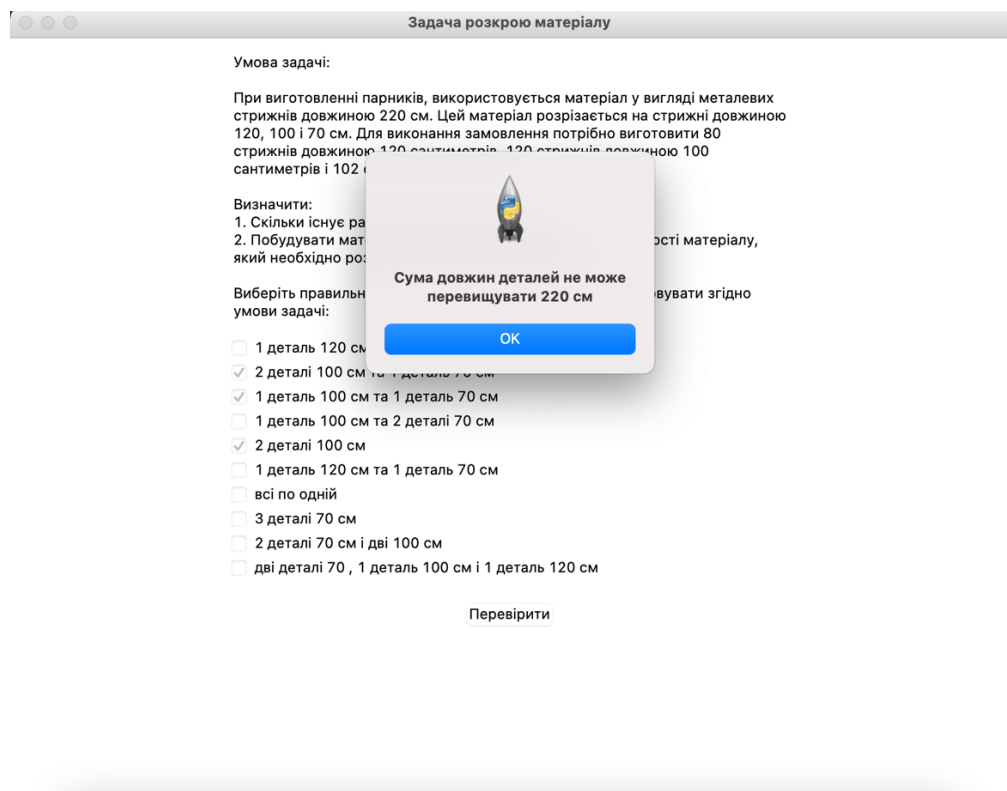


Рисунок 5.0 – Сповідження в разі хибної відповіді.

Для кожного завдання є своя підказка до кожної неправильної відповіді наприклад: (рис. 5.1)

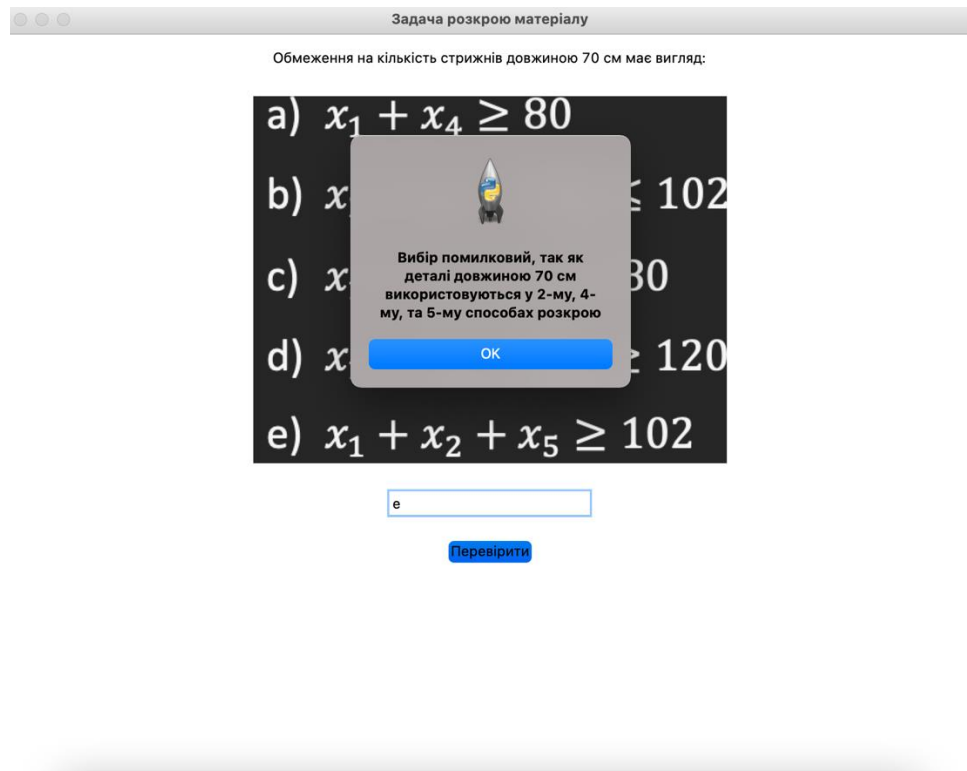


Рисунок 5.1 – Повідомлення про помилку.

Всі наступні питання після першого, виконані в формі письмової відповіді, тобто користувачу потрібно самому ввести варіант відповіді. (рис. 5.2)

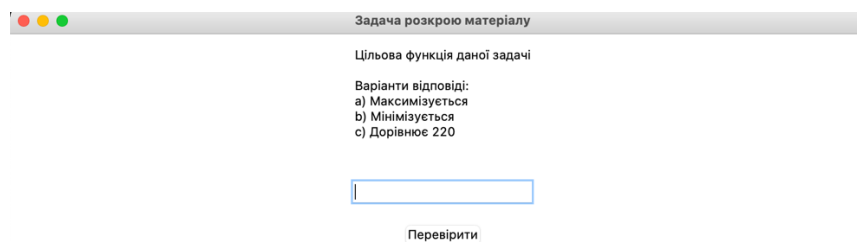


Рисунок 5.2 – Користувач повинен надрукувати варіант відповіді.

У разі неправильної відповіді користувача, тренажер відображає наступне попередження. (рис. 5.3)

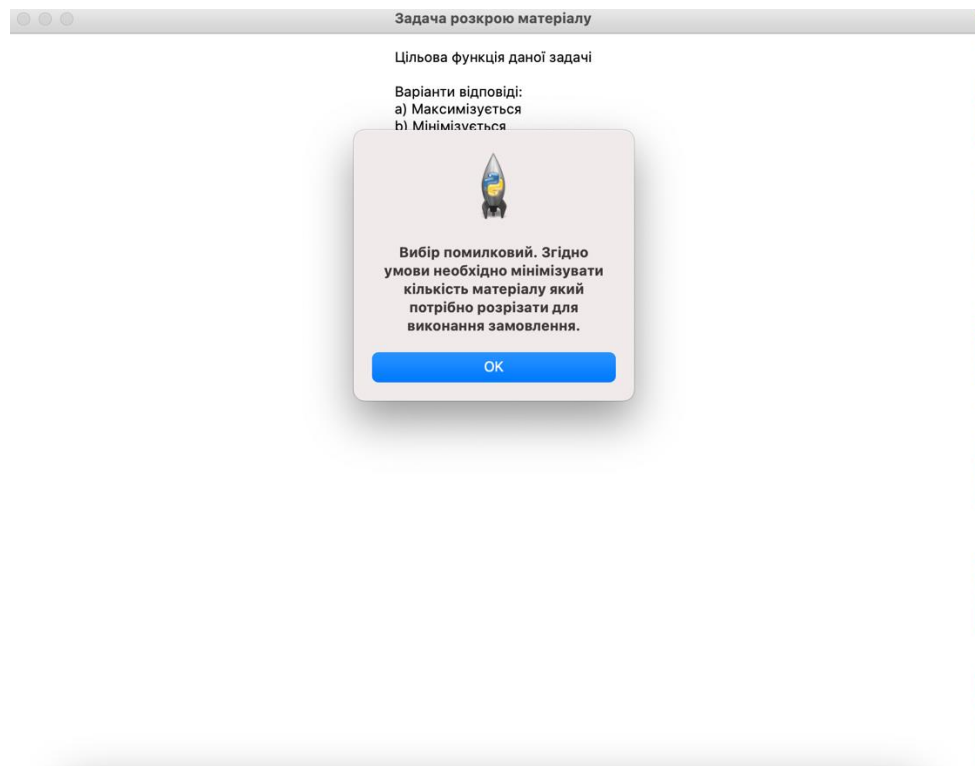


Рисунок 5.3 – Сповіщення в разі хибної відповіді.

Також в завданнях певного типу реалізовано кнопку “Способи розкрою” яка виглядає так(рис. 5.4)

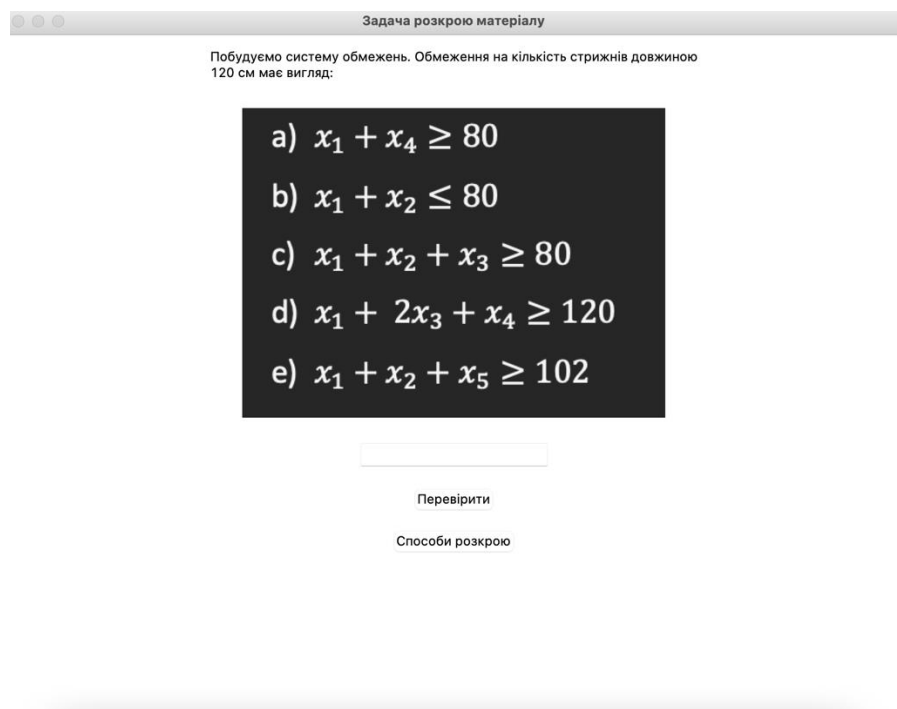


Рисунок 5.4 – Кнопка “способи розкрою”.

Після того, як користувач натиснув на кнопку “способи розкрою на екрані з’явиться повідомлення про способи розкрою”. (рис. 5.5)

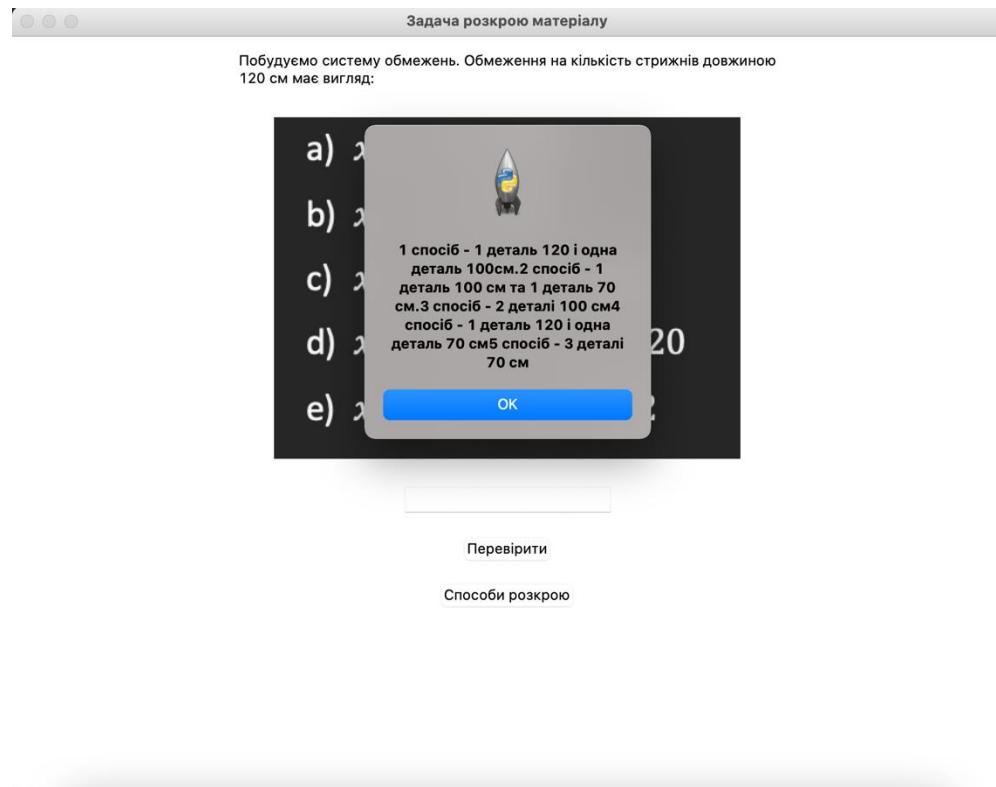


Рисунок 5.5 – Повідомлення у випадку натискання кнопки “способи розкрою”.

У разі неправильної відповіді користувача, система відображає наступне попередження. (рис. 5.6)

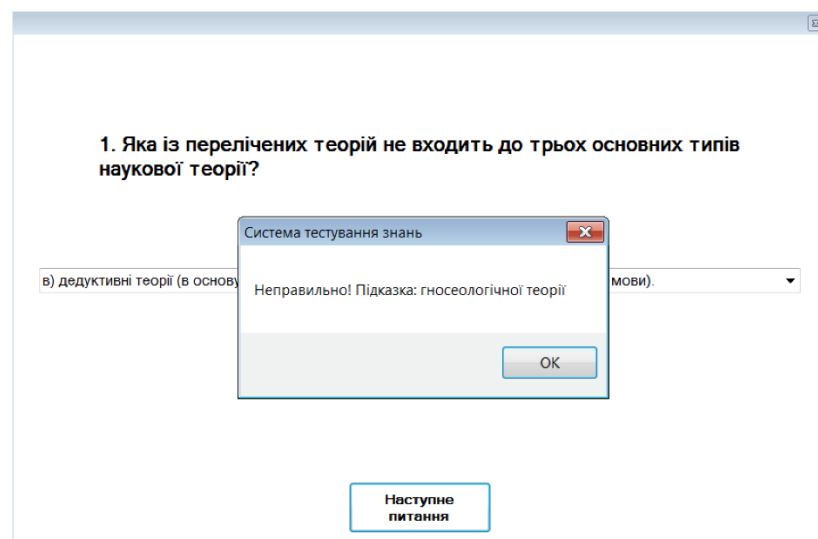


Рисунок 5.6 – Сповідження в разі хибної відповіді.

Після закінчення тестування, користувача вітає кінцеве вікно яке підводить підсумки. (рис. 5.7)

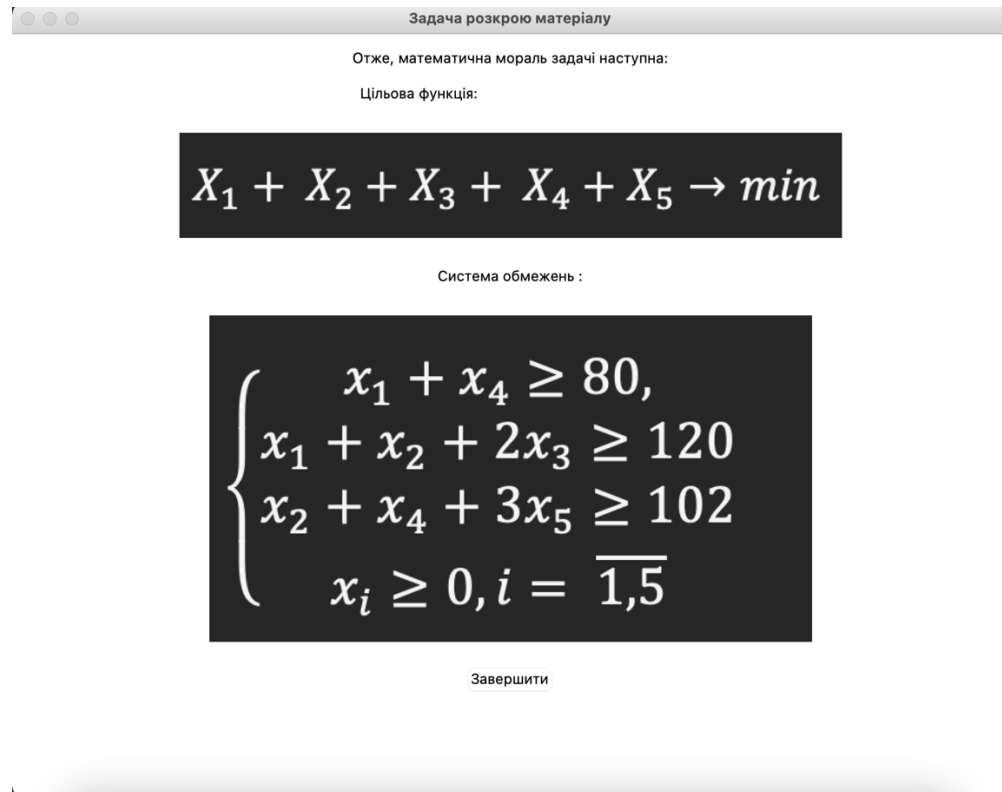


Рисунок 5.7 – Кінцеве вікно.

Після того, як користувач натиснув на кнопку завершити, програма завершить свою роботу. (рис. 5.8)

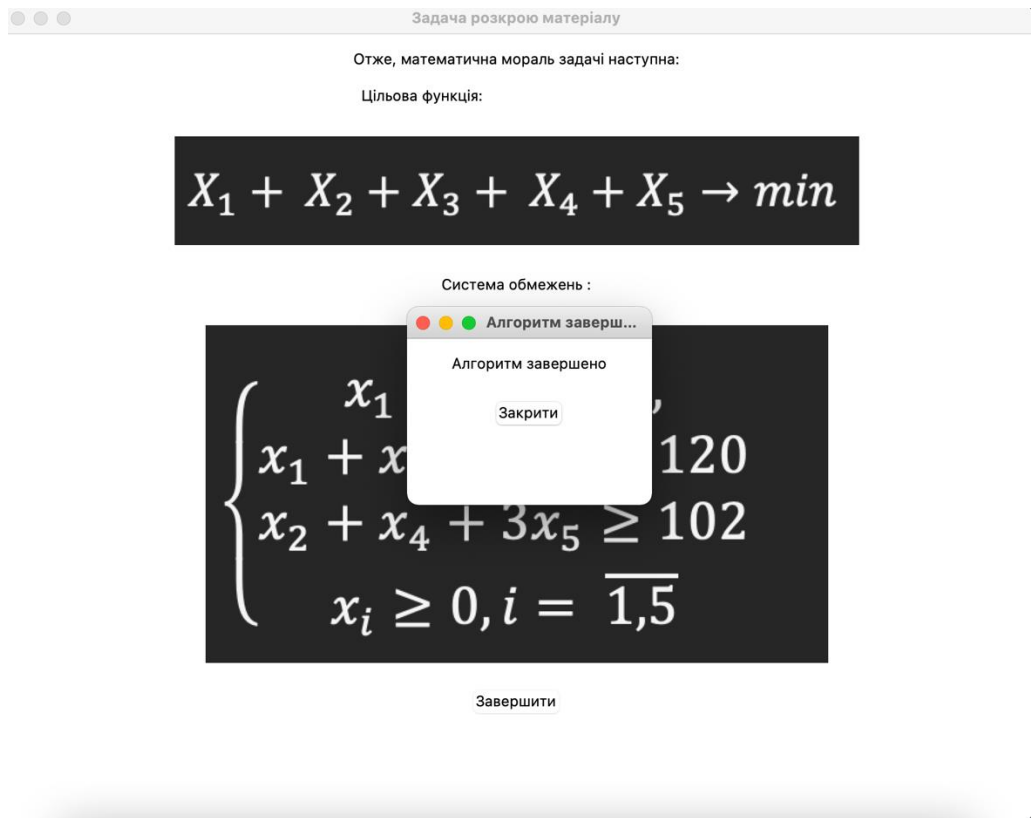


Рисунок 5.8 – завершення роботи програми

ВИСНОВКИ

Отже методи оптимізації та дослідження операцій (ОП) є ключовими інструментами для вирішення складних задач прийняття рішень у різних галузях, таких як інженерія, економіка, управління, логістика та фінанси. Їх використання дозволяє знаходити найкращі можливі рішення, максимізуючи або мінімізуючи певні цільові функції при дотриманні заданих обмежень.

По-перше, дослідження операцій застосовують математичні моделі, статистичний аналіз та алгоритмічні методи для аналізу та оптимізації складних систем. Це дає змогу організаціям підвищувати ефективність, зменшувати витрати та покращувати якість послуг або продукції.

По-друге, методи оптимізації та ОП включають широкий спектр підходів, таких як лінійне та нелінійне програмування, динамічне програмування, теорія ігор, теорія черг та евристичні алгоритми. Це дозволяє адаптувати методи до конкретних потреб і задач, що постають перед організаціями.

По-третє, ці методи стали невід'ємною частиною сучасного менеджменту та стратегічного планування, забезпечуючи інструменти для аналізу, моделювання та вирішення проблем, які раніше здавалися непосильними.

Отже, завдяки постійному розвитку комп'ютерних технологій та алгоритмів, методи оптимізації та дослідження операцій продовжують вдосконалюватися, відкриваючи нові можливості для ефективного управління та прийняття рішень у різних галузях. Вони залишаються важливим інструментом для досягнення високих результатів у сучасному світі.

Основні результати кваліфікаційної роботи:

1. Розроблено алгоритм та програмна реалізація тренажеру для побудови математичної моделі оптимального розкрою матеріалів

2. Розроблено вікна, що включають тестові завдання та питання з теми тестування, на які користувач повинен надавати вірні або невірні відповіді.

3. Створено вікно, зі стартовою кнопкою.

4. Створено кнопку, завдяки якій можна отримати інформацію по розкрою матеріалу.

5. Додано випадкову подачу питань користувачеві.

Тренажер з дистанційного курсу «Методи оптимізації та дослідження операцій» повністю готовий для використання та навчання.

1. Черненко О. О. Курсовий проєкт із фаху: методичні рекомендації щодо оформлення пояснювальних записок до курсового проєкту для студентів спеціальності 122 Комп'ютерні науки освітня програма «Комп'ютерні науки» ступеня бакалавра, магістра / О. О. Черненко. – Полтава : ПУЕТ, 2022. – 58 с. – 1 електрон. опт. диск (CVD-ROM).
2. Дистанційне навчання [Електронний ресурс]. – Режим доступу: <https://life.pravda.com.ua/society/2020/07/2/241517/>
3. Hillier, F. S., & Lieberman, G. J. (2020). *Introduction to Operations Research*. McGraw-Hill Education.
4. Cormen, T. H., Leiserson, C. E., Rivest, R. L., & Stein, C. (2009). *Introduction to Algorithms*. MIT Press
5. INFORMS (Institute for Operations Research and the Management Sciences) – Посилання: <https://www.informs.org/>
6. Wikipedia статті - https://uk.wikipedia.org/wiki/%D0%94%D0%BE%D1%81%D0%BB%D1%96%D0%B4%D0%B6%D0%B5%D0%BD%D0%BD%D1%8F_%D0%BE%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D1%96%D0%B9
7. SciPy – Посилання: <https://scipy.org/>
8. PuLP – Посилання: https://docs.pulpproject.org/pulp_python/
9. MATLAB Optimization Toolbox - <https://www.mathworks.com/products/optimization.html>

ДОДАТОК А. КОД ПРОГРАМИ