НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ

«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Кафедра автоматизованих систем обробки інформації та управління

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(повна назва кафедри, циклової комісії)

**КУРСОВА РОБОТА**

з                                           «Основи програмування»

(назва дисципліни)

на тему:                «Розвязання СЛАР наближеними методами»

Студента I курсу ІП-01 групи

напряму підготовки 6.050103 «Програмна інженерія»

спеціальності «Програмне забезпечення систем»

Галько М. В.

(прізвище та ініціали)

Керівник Муха І.П

Доцент кафедри АСОІУ

(посада, вчене звання, науковий ступінь, прізвище та ініціали)

Національна оцінка \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Кількість балів: \_\_\_\_\_\_\_\_\_\_Оцінка: ECTS \_\_\_\_\_

|  |  |  |  |
| --- | --- | --- | --- |
| Члени комісії |  |  |  |
|  | (підпис) |  | (вчене звання, науковий ступінь, прізвище та ініціали) |
|  |  |  |  |
|  | (підпис) |  | (вчене звання, науковий ступінь, прізвище та ініціали) |

Київ ‑ 2021 рік

Національний технічний університет України “КПІ”

(назва вищого навчального закладу)

Кафедра автоматизованих систем обробки інформації і управління

Дисципліна Основи програмування

Напрям "Програмна інженерія"

Курс І Група ІП-01 Семестр 2

**ЗАВДАННЯ**

**на курсову роботу студента**

|  |
| --- |
| Галько Міли Вячеславівни |

(прізвище, ім’я, по батькові)

|  |  |
| --- | --- |
| 1. Тема роботи | Розв'язання СЛАР наближеними методами |
|  | |
|  | |

|  |  |
| --- | --- |
| 2. Строк здачі студентом закінченої роботи |  |

|  |  |
| --- | --- |
| 3. Вихідні дані до роботи | Технічне завдання (додаток А) |
|  | |
|  | |
|  | |

4. Зміст розрахунково-пояснювальної записки (перелік питань, які підлягають розробці)

|  |
| --- |
|  |
|  |
|  |
|  |

5. Перелік графічного матеріалу ( з точним зазначенням обов’язкових креслень )

|  |
| --- |
|  |
|  |
|  |
|  |

|  |  |
| --- | --- |
| 6. Дата видачі завдання |  |

КАЛЕНДАРНИЙ ПЛАН

|  |  |  |  |
| --- | --- | --- | --- |
| № п/п | Назва етапів курсової роботи | Термін виконання етапів роботи | Підписи керівника, студента |
| 1. | Отримання теми курсової роботи | 14.03.16-20.03.16 |  |
| 2. | Підготовка ТЗ | 21.03.16-03.04.16 |  |
| 3. | Пошук та вивчення літератури з питань курсової роботи | 04.04.16-17.04.16 |  |
| 4. | Розробка алгоритму вирішення задачі | 18.04.16-24.04.16 |  |
| 6. | Узгодження алгоритму з керівником | 18.04.16-24.04.16 |  |
| 5. | Розробка сценарію роботи програми | 25.04.16-01.05.16 |  |
| 6. | Узгодження сценарію роботи програми з керівником | 25.04.16-01.05.16 |  |
| 7. | Узгодження з керівником інтерфейсу користувача | 25.04.16-01.05.16 |  |
| 8. | Розробка програмного забезпечення | 02.05.16-22.05.16 |  |
| 9. | Налагодження розрахункової частини програми | 02.05.16-22.05.16 |  |
| 10. | Розробка та налагодження інтерфейсної частини програми | 23.05.16-29.05.16 |  |
| 11. | Узгодження з керівником набору тестів для контрольного прикладу | 23.05.16-29.05.16 |  |
| 12. | Тестування програми | 30.05.16-05.06.16 |  |
| 13. | Підготовка пояснювальної записки | 06.06.16-12.06.16 |  |
| 14. | Здача курсової роботи на перевірку | 06.06.16-12.06.16 |  |
| 15. | Захист курсової роботи |  |  |
|  |  |  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| Студент |  |  | Галько М. В. |
|  | (підпис) |  | (прізвище, ім’я, по батькові) |

|  |  |  |  |
| --- | --- | --- | --- |
| Керівник |  |  | Муха І. П. |
|  | (підпис) |  | (прізвище, ім’я, по батькові) |

"\_\_\_" \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ 2021 р.

Анотація

Пояснювальна записка до курсової роботи: 80 сторінок, 22 рисунків,   
8 таблиць, 10 посилань.

Об’єкт дослідження: задача знаходження розв’язків СЛАР.

Мета роботи: дослідження методів наближеного розв’язання СЛАР ітераційними методами, створення програмного забезпечення для швидкісного розв’язання надвеликих систем лінійних алгебраїчних рівнянь.

Вивчено методи швидкісного розв’язання СЛАР. Приведені змістовні постановки задач, їх індивідуальні математичні моделі, а також описано детальний процес розв’язання кожної з них.

Виконана програмна реалізація алгоритмів Якобі, Гауса-Зейделння, градієнтного спуску (спряжених градієнтів).

(Больше ключевых слов)

МЕТОД ЯКОБІ, МЕТОД ГАУСА-ЗЕЙДЕЛЯ, МЕТОД ГРАДІЄНТНОГО СПУСКУ, МЕТОД СПРЯЖЕНИХ ГРАДІЄНТІВ, РІШЕННЯ СЛАР ІТЕРАЦІЙНИМИ МЕТОДАМИ

Зміст

[Вступ 6](#_Toc451632576)

[1 Постановка задачі 7](#_Toc451632577)

[2 Теоретичні відомості 8](#_Toc451632578)

[2.1. Метод Якобі 8](#_Toc451632579)

[2.2. Метод Гауса-Зейделя 9](#_Toc451632580)

[2.3. Метод градієнтного спуску (спряжених градієнтів) 10](#_Toc451632581)

[3 Опис алгоритмів 13](#_Toc451632582)

[3.1. Загальний алгоритм 13](#_Toc451632583)

[3.2. Алгоритм методу Якобі 14](#_Toc451632584)

[3.3. Алгоритм методу Гауса-Зейделя 17](#_Toc451632585)

[3.4. Алгоритм методу градієнтного спуску 19](#_Toc451632586)

[4 Опис програмного забезпечення 21](#_Toc451632587)

[4.1. Функціональна структура програмного забезпечення 21](#_Toc451632588)

[4.2. Опис функцій частин програмного забезпечення 21](#_Toc451632589)

[4.2.1. Користувацькі функції 22](#_Toc451632590)

[4.2.2. Стандартні функції 31](#_Toc451632591)

[5 Тестування програмного забезпечення 33](#_Toc451632592)

[5.1. План тестування 33](#_Toc451632593)

[5.2. Приклади тестування 33](#_Toc451632594)

[5.2.1. Виконання програми, коли у матрицю системи невірно введено число 33](#_Toc451632595)

[5.2.2. Виконання програми, коли у стовпець вільних членів невірно введено число 34](#_Toc451632596)

[5.2.3. Виконання програми, коли у матрицю системи введено число з точністю більшою, ніж 3 знаки після коми 35](#_Toc451632597)

[5.2.4. Перевірка вірності роботи програми, коли користувачем вірно введені всі дані та для обраного методу рішення сходиться 36](#_Toc451632598)

[6 Інструкція користувача 38](#_Toc451632599)

[6.1. Робота з програмою 38](#_Toc451632600)

[6.2. Формат вхідних та вихідних даних 43](#_Toc451632601)

[6.3. Системні вимоги 43](#_Toc451632602)

[7 Аналіз і узагальнення результатів 45](#_Toc451632603)

[Висновки 50](#_Toc451632604)

[Перелік посилань 51](#_Toc451632605)

[**Додаток А Технічне завдання** 52](#_Toc451632606)

[Додаток Б Тексти програмного коду 55](#_Toc451632607)

Усі розділи, підрозділи, пункти, підпункти, перелік додатків (визначення сторінок необов’язкове + без посилання на «Зміст»)

Вступ

Чисельне рішення СЛАР – одна з найпоширеніших задач у науково-технічних дослідженнях. Така задача виникає в математичній фізиці (чисельне рішення диференціальних та інтегральних рівнянь), економіці, статистиці. При цьому прикладні задачі часто потребують вирішення великих і надвеликих СЛАР з числом невідомих більше 1000. До таких СЛАР, наприклад, призводить чисельне рішення двовимірних і особливо тривимірних задач математичної фізики.

Існуючі бібліотеки програм розроблені на основі, так званих, прямих методів розв'язку СЛАР (наприклад метод Гауса та його модифікації). Число арифметичних операцій таких методів для знаходження чисельного розв'язку СЛАР розмірністю   
складає – [1]. Кубічна залежність числа арифметичних операцій від розміру матриці СЛАР призводить до нереально великого часу рішення при навіть на найсучасніших ЕОМ.

Ітераційні методи розв'язку СЛАР набагато економніші, як по машинному часу рішення, так і по використанню оперативної пам'яті. Так, якщо ітераційний метод швидко сходяться з числом ітерацій , то час виконання, пропорційний вже квадрату розміру матриці – [3]. При використанні ітераційних методів, що швидко сходяться, стає можливим рішення СЛАР розмірністю .

В даний час відсутні бібліотеки та програми широкого призначення для чисельного рішення великих і надвеликих СЛАР. Таким чином, розробка ефективних ітераційних алгоритмів для рішення СЛАР великої розмірності є актуальним завданням.

Тому метою даної роботи є вивчення та аналіз роботи найбільш відомих розвязання СЛАР, таких як метод простих ітерацій (або так званий метод Якобі), метод Гауса-Зейделя, метод градієнтного спуску (або спряжених градієнтів).

# Постановка задачі

призначення:

- ціль роботи

- сутність вирішення

Основна частина

Аналіз предметної області

Опис архітектури програмної системи

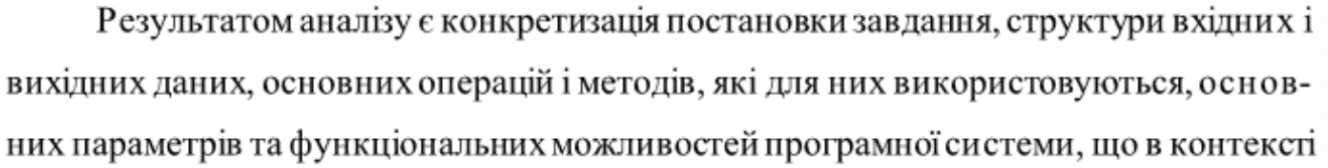
Опис програмного забезпечення

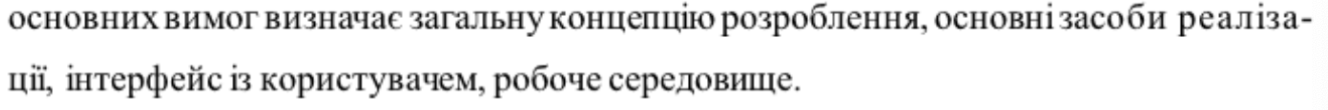
Результати його тестування

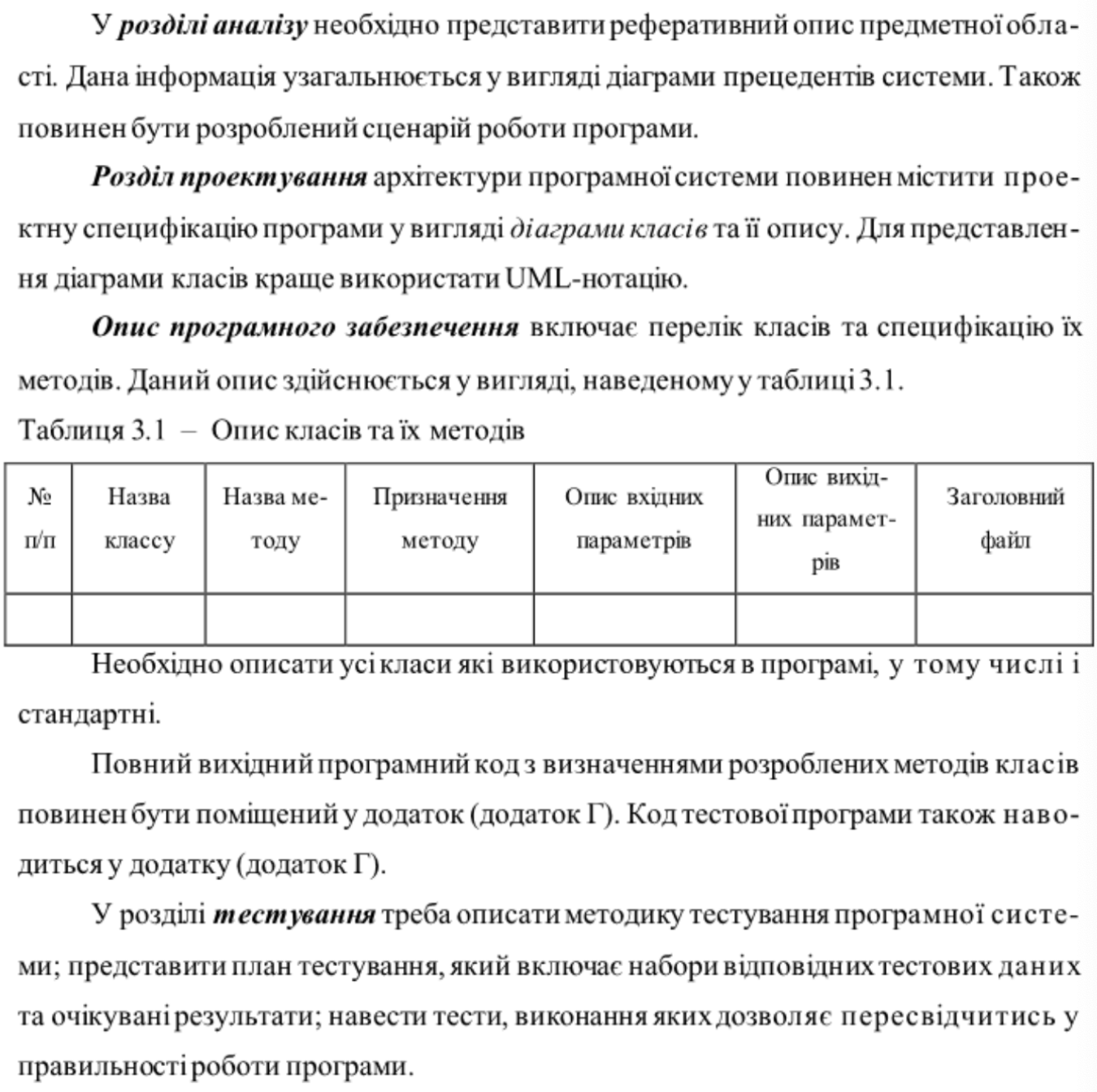
Інструкція користувача

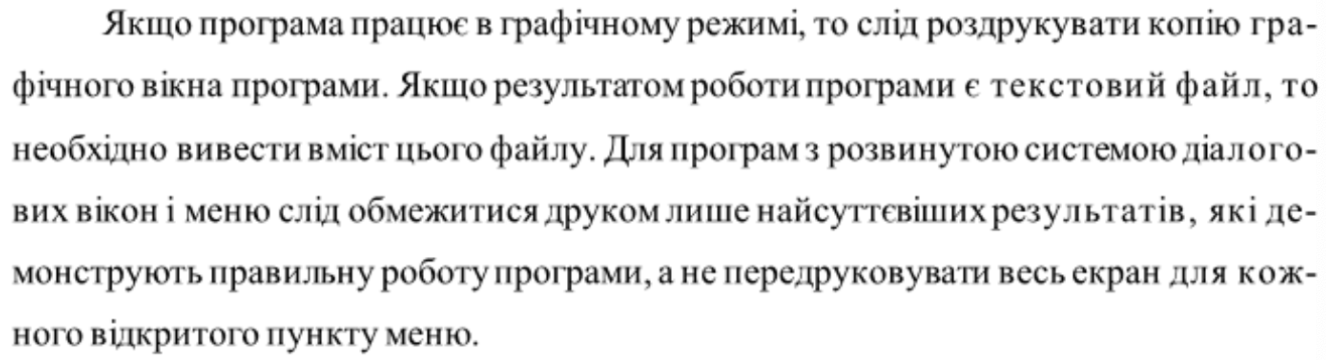
Постановка задачі

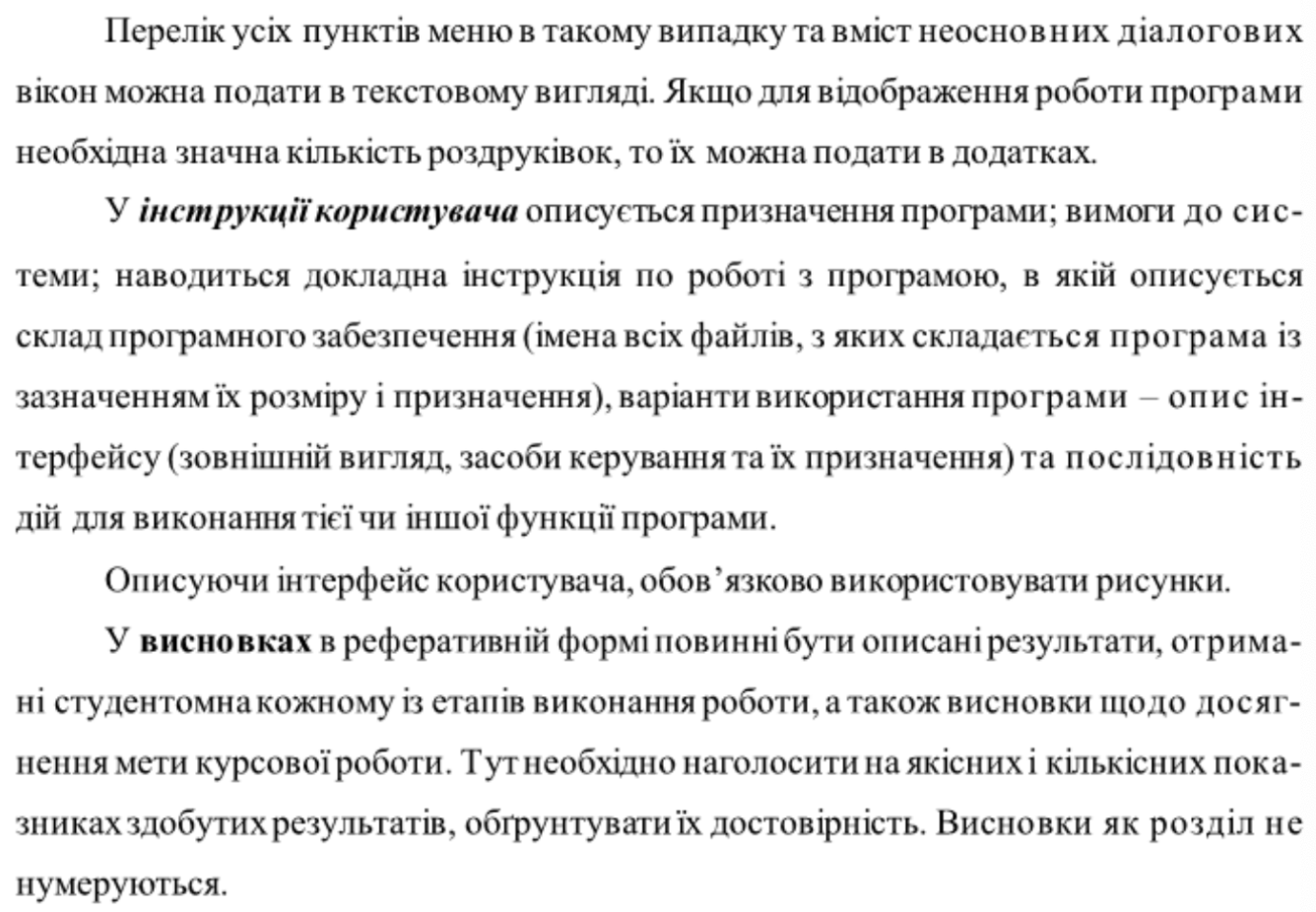
Додаток А



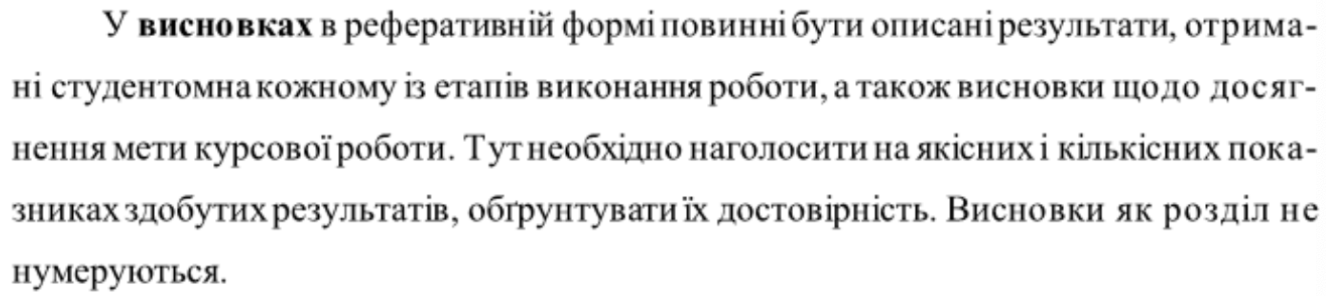




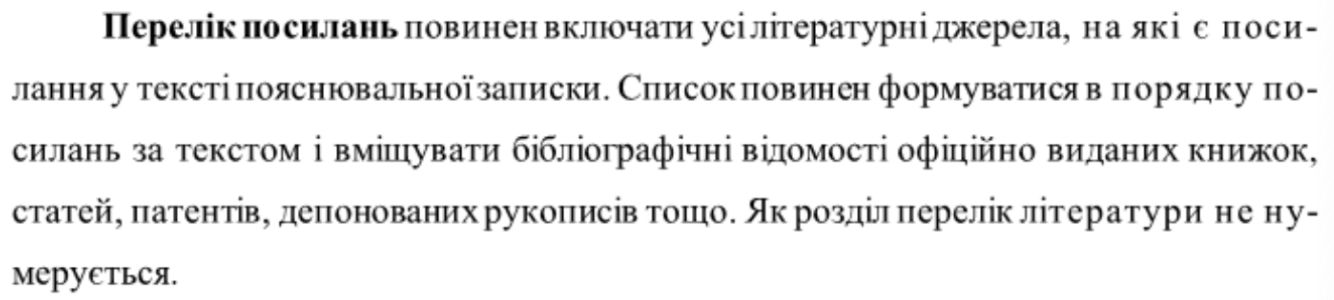




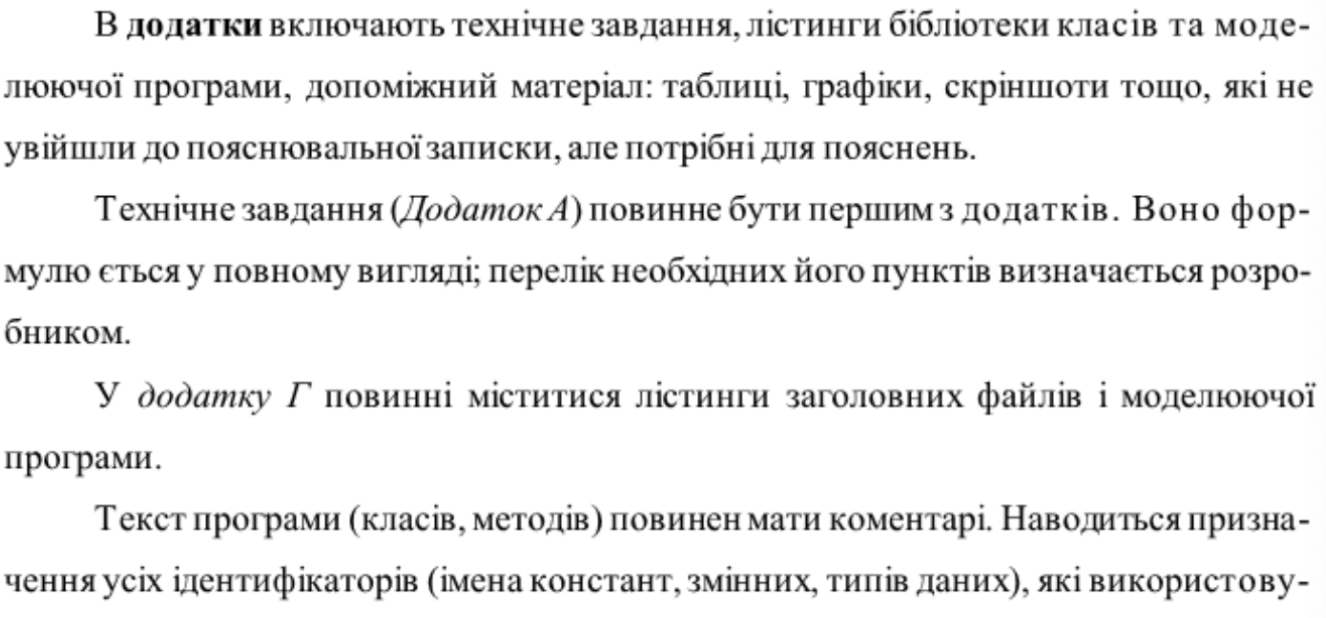
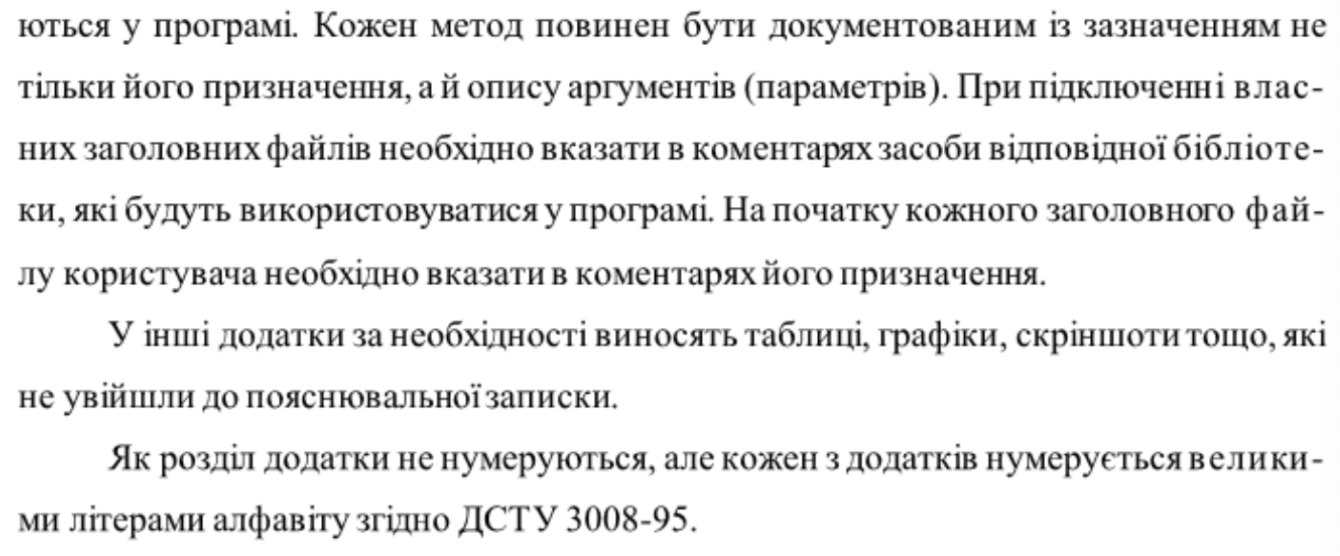
Висновок



Перелік посилань



Додатки

Тексти програмного коду програмного забезпечення

<Тема курсової роботи>

\_\_\_\_\_\_(Найменування програми (документа))\_\_\_\_\_\_\_\_

\_\_\_\_\_\_CD-RW\_\_\_\_\_\_\_

(Вид носія даних)

\_\_\_\_арк, Кб\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(Обсяг програми (документа), арк., Кб)

студента I групи ІП-01

<ПІБ>

Технічне завдання

КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ ім. І. Сікорського

Кафедра

автоматизованих систем обробки інформації та управління

Затвердив

Керівник \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

«\_\_\_»\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_201\_ р.

Виконавець:

Студент\_*Галько Міла Вячеславівна\_*

«\_\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_2021 р.

ТЕХНІЧНЕ ЗАВДАННЯ

на виконання курсової роботи

на тему: "Розв'язання СЛАР наближеними методами"

з дисципліни:

«Основи програмування»

Київ 2021

*Мета*: Метою курсової роботи є розробка програми, що розв’язує систему лінійних рівнянь, вигляд якої визначається керівником, наближеними методами:

метод простої ітерації (Якобі);

метод Гауса-Зейделя;

метод найшвидшого спуску (градієнта);

*Дата початку роботи*: «8» березня 2021 р.

*Дата закінчення роботи*: «\_\_\_»\_\_\_\_\_\_\_\_\_\_\_ 202\_ р

*Вимоги до програмного забезпечення*.

Функціональні вимоги:

Можливість розв’язувати СЛАР методами: простої ітерації (Якобі), Гауса-Зейделя та найшвидшого спуску (градієнта).

Можливість введення СЛАР вручну.

Можливість генерування СЛАР випадковим чином.

Можливість обирати метод розв’язку СЛАР.

На вхід подається: матриця arrC (коефіцієнти системи), матриця arrX (невідомі) та матриця arrF (вільні члени).

На виході програма повертає результат: невідомі та їх значення або СЛАР є несумісною.

Нефункціональні вимоги:

Програма написана на мові С++.

Користувач взаємодіє з програмою через графічну форму.

Програма дає можливість користувачеві ввести СЛАР через інтерфейс програми.

Програма має перевіряти введення користувача та запросити повторне введення при некоректності запропонованих даних.

Програма відображає основні кроки розв’язку СЛАР згідно з обраним методом, але лише у випадку сумісності системи.

Програма відображає кінцевий результат або повідомляє, що запропонована СЛАР є несумісною.

Все програмне забезпечення та супроводжуюча технічна документація повинні задовольняти наступним ДЕСТам:

ГОСТ 29.401 - 78 - Текст програми. Вимоги до змісту та оформлення.

ГОСТ 19.106 - 78 - Вимоги до програмної документації.

ГОСТ 7.1 - 84 та ДСТУ 3008 - 95 - Розробка технічної документації.

*Стадії та етапи розробки*:

Об'єктно-орієнтований аналіз предметної області задачі (до\_\_.\_\_.202\_ р.)

Об'єктно-орієнтоване проектування архітектури програмної системи (до \_\_.\_\_.202\_р.)

Розробка програмного забезпечення (до \_\_.\_\_.202\_р.)

Тестування розробленої програми (до \_\_.\_\_.202\_р.)

Розробка пояснювальної записки (до \_\_.\_\_.202\_ р.).

Захист курсової роботи (до \_\_.\_\_.202\_ р.).

*Порядок контролю та приймання*. Поточні результати роботи над КР регулярно демонструються викладачу. Своєчасність виконання основних етапів графіку підготовки роботи впливає на оцінку за КР відповідно до критеріїв оцінювання.

Для мене

**Тема:** СЛАР наближеними методами (загальний вигляд системи визначається керівником). Методи: простої ітерації Якобі, Гауса-Зейделя, найшвидшого спуску (градієнта).

**Розробити** програму у відповідності до завдання та передбачити:

1) введення вихідних даних за допомогою розробленої самостійно екранної форми (введення частини даних організувати засобами меню)(графічний інтерфейс);

2) збереження отриманих числових результатів у вигляді текстового файлу;

3) за можливості графічне представлення одержаних результатів;

4) демонстрацію (за можливості графічну) роботи застосованих методів (алгоритмів) розв’язання підзадач;

5) оцінку складності алгоритмів;

6) порівняльний аналіз методів (алгоритмів) розв’язання підзадач;

**Етапи:**

