Міністерство освіти і науки України

Сумський державний університет

Кафедра комп’ютерних наук

Секція інформаційно-комунікаційних технологій

**Пояснювальна записка**

**до курсової роботи**

з дисципліни

«Програмування»

Викладач Прокопенко В.М

Студент Іващенко Н.Ю

Група КН-42/2

Варіант Варіант №1

Суми – 2025

**Зміст**

[1. Постановка задачі 3](#_Toc198476310)

[2. Теоретичний матеріал з теми 4](#_Toc198476311)

[3. Опис структури даних та вимог до них 5](#_Toc198476312)

[4. Алгоритм роботи програми 8](#_Toc198476313)

[5. Опис функції користувача 9](#_Toc198476314)

[6. Опис файлів та їх призначення 11](#_Toc198476315)

[7. Список використаних бібліотек 12](#_Toc198476316)

[8. Інструкція для роботи з програмою 13](#_Toc198476317)

[**8.1 Необхідні ресурси для запуску програми 13**](#_Toc198476318)

[**8.2 Що необхідно для запуску програми. 13**](#_Toc198476319)

[**8.3 Як відповідати на запити програми? 14**](#_Toc198476320)

[**8.4 Перевірка та відловлювання помилок 14**](#_Toc198476321)

[9 Приклад тестування та результат роботи програми 15](#_Toc198476322)

[10 Графіки 17](#_Toc198476323)

[11 Висновки 18](#_Toc198476324)

[12 Список використаної літератури 19](#_Toc198476325)

1. **Постановка задачі**

Варіант 1

Описати масив структур із трьох елементів. Кожна структура об’єднує дані для одного варіанта розрахунку. Необхідно для кожного варіанта на відрізку часу від 0 до Т із кроком Δt побудувати графік зміни потужності N1 , що витрачається на розбризкування рідини осесиметричним розбризкувачем (рис. 1. 1) [2].

A screenshot of a math problem

AI-generated content may be incorrect.

Рисунок 1. 1 - Постановка задачі

1. **Теоретичний матеріал з теми**

C++ — це мова системного та об'єктно-орієнтованого програмування, що поєднує високу швидкодію з гнучкими можливостями проєктування. Вона дозволяє працювати як на низькому рівні з пам'яттю та ресурсами, так і створювати складні абстракції, моделі, алгоритми та великі програмні системи. C++ застосовується там, де потрібна ефективність, контроль і масштабованість: в ігровій індустрії, вбудованих системах, фінансових розрахунках, розробці драйверів та інженерному програмуванні [3].

Основні визначення, що використовувалися в програмі

* Структура Variant — це агрегований тип даних, який використовується для зберігання параметрів одного варіанта вхідних даних.
* Omega1(t) — функція, що обчислює миттєву кутову швидкість за допомогою гармонічної модифікації.
* H\_t(t) — функція, що визначає висоту в залежності від часу на основі кусочно-заданої функції.
* N1(t) — функція, що обчислює потужність у певний момент часу на основі вхідних параметрів варіанта.
* Константа g = 9.81 — прискорення вільного падіння.

Функції:

* cos(x) — обчислює косинус (використовується в Omega1).
* pow(x, y) — підносить x до степеня y.
* isnan(x) — перевіряє, чи є результат обчислення невизначеним (NaN).
* std::fixed + std::setprecision(n) — встановлює формат виводу з фіксованою кількістю знаків після коми.
* Оператори вводу/виводу (>>, <<) — використовуються для зчитування/виведення даних у потоки.

1. **Опис структури даних та вимог до них**

Таблиця опису структури даних(табл 3. 1).

|  |  |  |  |
| --- | --- | --- | --- |
| **Ім’я параметра у формулі** | Змінна у програмі | Тип змінної | Призначення |
| ρ (густина) | rho | double | Вхідний параметр, густина рідини в системі. (зчитується з файлу input.txt) |
| h₀ (початкова висота) | h0 | double | Вхідний параметр, базова висота струменя. (зчитується з файлу input.txt) |
| β (амплітуда коливань висоти) | beta | double | Вхідний параметр, визначає зміну висоти в часі.  (зчитується з файлу input.txt) |
| μ (коефіцієнт витрати) | mu | double | Вхідний параметр, впливає на витрату.  (зчитується з файлу input.txt) |
| a | a | double | Вхідний параметр, коефіцієнт площі чи масштабу  (зчитується з файлу input.txt). |
| S (площа поперечного перерізу) | S | double | Вхідний параметр, площа отвора або труби  (зчитується з файлу input.txt). |
| T (період) | T | double | Вхідний параметр, повний період коливань.  (зчитується з файлу input.txt) |
| Продовження таблиці 3.1 | | | |
| Δ (крок зміни часу) | Delta | double | Вхідний параметр, використовується для дискретизації часу.  (зчитується з файлу input.txt) |
| Ω₀ (середня кутова швидкість) | Omega0 | double | Вхідний параметр, базове значення кутової швидкості.  (зчитується з файлу input.txt) |
| α (амплітуда кутової швидкості) | alpha | double | Вхідний параметр, визначає зміну кутової швидкості.  (зчитується з файлу input.txt) |
| f\_вих (вихідна площа) | f\_vyh | double | Вхідний параметр, площа вихідного отвору.  (зчитується з файлу input.txt) |
| R | R | double | Вхідний параметр, геометричний розмір.  (зчитується з файлу input.txt) |
| R₁ | R1 | double | Вхідний параметр, геометричний розмір, використовується в розрахунках.  (зчитується з файлу input.txt) |
| g (прискорення вільного падіння) | g | const double | Константа, гравітаційне прискорення (9.81 м/с²). |
| Продовження таблиці 3.1 | | | |
| t | t | double | Змінна циклу, час, у якому відбувається обчислення. (записується у файл output.txt) |
| H(t) | H (через H\_t) | double | Проміжна змінна, визначає висоту струменя у момент часу. |
| Ω(t) | omega (через Omega1) | double | Проміжна змінна, кутова швидкість у момент часу. |
| N₁(t) | n1 | double | Результат, значення потужності на момент часу t.  (записується у файл output.txt) |

Таблиця 3. 1 - Опис структури даних

1. **Алгоритм роботи програми**

Відкрити файл

data/input.txt

Чи відкрився файл?

Зчитування параметрів для 3-х варіантів у масив variants[3]

Цикл по кожному варіанту (i = 0 до 2)

Так

Розрахунок Ω(t) та H(t)

Обчислення потужності N1(t)

a == 0, S == 0, підкореневий вираз < 0

Старт

Так

Вивести результат у консоль. Записати у файл..

Кінець

Ні

Виведення про помилку

Виведення про помилку

Ні

1. **Опис функції користувача**

У процесі виконання курсової роботи було реалізовано декілька користувацьких функцій, призначених для обчислення фізичних параметрів на основі вхідних даних. Нижче подано опис кожної функції, її призначення, а також перелік вхідних і вихідних параметрів.

1. double Omega1(double t, double Omega0, double alpha, double T)

Призначення:  
Обчислює миттєве значення кутової швидкості обертання тіла в момент часу t за заданим законом коливань.

Вхідні параметри:

* t (double) — поточний момент часу, с.
* Omega0 (double) — середнє значення кутової швидкості, рад/с.
* alpha (double) — амплітуда відносного коливання кутової швидкості.
* T (double) — період коливань, с.

Вихідні дані:

* Кутова швидкість Omega1(t) (double) у момент часу t, рад/с.

2. double H\_t(double t, double h0, double beta, double T)

Призначення:  
Обчислює висоту рідини або рівня в певний момент часу t відповідно до трифазного коливального процесу.

Вхідні параметри:

* t (double) — поточний момент часу, с.
* h0 (double) — середня висота рідини, м.
* beta (double) — відносна амплітуда коливань висоти.
* T (double) — період коливань, с.

Вихідні дані:

* Висота H(t) (double) у момент часу t, м.

3. double N1(double t, const Variant& v, double g = 9.81)

Призначення:  
Обчислює потужність N1 у момент часу t на основі фізичних параметрів конструкції та динаміки процесу.

Вхідні параметри:

* t (double) — момент часу, с.
* v (Variant) — структура з параметрами варіанту:
  + ho (double) — середня висота рідин, м
* h0, beta, mu, a, S, T, Delta, Omega0, alpha, f\_vyh, R, R1 (double) — фізичні та геометричні параметри, детальніше описані в розділі вхідних даних.
* g (double) — прискорення вільного падіння, м/с² (за замовчуванням — 9.81).

Вихідні дані:

* Потужність N1(t) (double) у ватах. У разі помилки (наприклад, недопустимі значення) повертає NaN і виводить повідомлення про помилку.

1. **Опис файлів та їх призначення**

1. main.cpp

Призначення:

Основний програмний файл, що містить:

* оголошення структури Variant для зберігання вхідних параметрів;
* реалізацію користувацьких функцій (Omega1, H\_t, N1);
* зчитування вхідних даних з файлу;
* обчислення потужності у задані моменти часу;
* виведення результатів у консоль та у вихідний файл.

Цей файл є головним модулем програми, з якого починається виконання.

2. input.txt

Призначення:

Вхідний текстовий файл, у якому зберігаються вхідні параметри для розрахунку трьох варіантів задачі. Кожен рядок містить набір параметрів, що відповідає одному варіанту, у наступному порядку

rho h0 beta mu a S T Delta Omega0 alpha f\_vyh R R1

3. output.txt

Призначення:

Вихідний текстовий файл, у який записуються результати обчислень потужності N1 для кожного варіанта в задані моменти часу t. Результати подаються у форматі:

=== Variant N ===

t = ... c, N1 = ... watt

У разі помилкових обчислень (наприклад, ділення на нуль або корінь з від’ємного числа), виводиться повідомлення error (NaN).

1. **Список використаних бібліотек**

1. <iostream>

Призначення:  
Забезпечує введення/виведення даних через консоль за допомогою об'єктів std::cin, std::cout, std::cerr [3].

2. <fstream>

Призначення:  
Надає засоби для роботи з файлами. У даній програмі використовується для:

* зчитування вхідних даних з файлу (std::ifstream),
* запису результатів у вихідний файл (std::ofstream) [3]

3. <cmath>

Призначення:  
Містить математичні функції. У програмі використовуються:

* cos() — обчислення косинуса,
* pow() — піднесення до степеня,
* sqrt() (опосередковано через pow(..., 1.5)),
* M\_PI — математична константа π (якщо не визначена, задається вручну) [3].

4. <vector>

Призначення:  
Надає можливість використовувати динамічні масиви. У програмі застосовується для зберігання кількох варіантів вхідних даних (std::vector<Variant>) [1].

5. <iomanip>

Призначення:  
Забезпечує форматований вивід чисел. Використовується для встановлення кількості знаків після коми (std::setprecision) і фіксованого формату виводу (std::fixed) [1].

1. **Інструкція для роботи з програмою**

## **8.1 Необхідні ресурси для запуску програми**

Файл з вхідними даними data/input.txt

Має містити 3 варіанти вхідних даних у форматі (1 рядок = 1 варіант).

rho h0 beta mu a S T Delta Omega0 alpha f\_vyh R R1 (1 варіант)

rho h0 beta mu a S T Delta Omega0 alpha f\_vyh R R1 (2 варіант)

rho h0 beta mu a S T Delta Omega0 alpha f\_vyh R R1 (3 варіант)

Приклад:

1000 0.3 0.3 0.4 3 0.4 500 25 314.159 0.1 1.1e-5 0.15 0.1 (1 варіант)

1050 0.25 0.2 0.3 2.5 0.3 500 25 345.575 0.15 1.0e-5 0.2 0.15 (2 варіант)

1100 0.2 0.1 0.25 2.2 0.25 500 25 376.991 0.2 0.9e-5 0.25 0.2 (3 варіант)

Середовище компіляції**:** будь-який компілятор C++ з підтримкою C++11 і вище( наприклад наприклад: g++, clang++, або IDE Clion, Visual Studio).

## **8.2 Що необхідно для запуску програми.**

Якщо це IDE:

* Visual Studio: Відкрити головний файл проекту Main.cpp у Visual Studio. Натиснути CTRL + F5. Результати обчислень перевірити на екрані консолі, та у файлі output.txt.
* Clion: Відкрити головний файл проекту Main.cpp у Clion. Натиснути SHIFT + F10. Результати обчислень перевірити на екрані консолі, та у файлі output.txt.

Якщо це термінал/файловий менеджер:

На лінукс:

* Відкрити термінал (зазвичай комбінація клавіш CTRL + T)
* Ввести команду sudo nano /data/input.txt
* Ввести валідні дані (числові значення) згідно заданого порядку ( 1 рядок = 1 варіант, кожна змінна розділена пробілом)
* Натиснути Ctrl + O для збереження змін у файлі
* Натиснути Ctrl + X для виходу з файлу
* Скомпілювати програму g++ main.cpp -o run , та запустити ./run

На Windows:

* Через стандартний файловий менеджер перейти до директорії data/input.txt
* Відкрити файл input.txt через додаток “Блокнот”
* Ввести валідні дані (числові значення) згідно заданого порядку ( 1 рядок = 1 варіант, кожна змінна розділена пробілом)
* Зберегти зміни натиснувши CTRL + S. Вийти з блокноту
* Відкривши термінал, скомпілювати програму завдяки команді g++ main.cpp -o run
* Запустити програму за допомогою команди ./run
  1. **Як відповідати на запити програми?**

Програма сама не просить у користувача вводити дані з клавіатури — всі запити читаються з input.txt. Проте потрібно:

* Впевнитися, що всі параметри є і валідні (числові значення).
* Перевірити, що значення параметрів не призводять до математичних помилок (наприклад: ділення на нуль, від’ємний підкореневий вираз).
  1. **Перевірка та відловлювання помилок**

Програма виконує такі перевірки:

* Якщо параметри призводять до base < 0, або a == 0, або S == 0 — обчислення не виконується, виводиться error (NaN):

if (base < 0 || v.a == 0 || v.S == 0) {

std::cerr << «Error when t = « << t << «: invalid values (base < 0 or / 0)\n»;

return NAN;

}

* Такі помилки виводяться у консоль та у файл output.txt.

1. **Приклад тестування та результат роботи програми**

Вхідні дані(input.txt):

1000 0.3 0.3 0.4 3 0.4 500 25 314.159 0.1 1.1e-5 0.15 0.1

1050 0.25 0.2 0.3 2.5 0.3 500 25 345.575 0.15 1.0e-5 0.2 0.15

1100 0.2 0.1 0.25 2.2 0.25 500 25 376.991 0.2 0.9e-5 0.25 0.2

Результат роботи програми(консоль, рис 9. 1):

A screenshot of a computer program

AI-generated content may be incorrect.

Рисунок 9. 1 - Результат роботи програми(при валідних даних)

Вхідні дані(input.txt):

1000 0.3 0.3 0.4 3 0.4 500 25 314.159 0.1 1.1e-5 0.15 0.1

1050 0.25 0.2 0.3 2.5 0.3 500 25 345.575 0.15 1.0e-5 0.2 0.15

text 0.2 txe 0.25 2.2 text 500 25 tex tx 0.9e-5 0.25 0.2

Результат роботи програми(консоль, рис 9. 2):

A screenshot of a computer program

AI-generated content may be incorrect.

Рисунок 9. 2 - Результат роботи програми(при не валідних даних)

1. **Графіки**

Нижче представлений графік відображаючий різницю розрахунків для трьох різних варіантів (Зелений = 3 ваіріант, Жовтий = 2 варіант, Помаранчевий = 1 варіант).

Вхідні дані:

1000 0.3 0.3 0.4 3 0.4 500 25 314.159 0.1 1.1e-5 0.15 0.1

1050 0.25 0.2 0.3 2.5 0.3 500 25 345.575 0.15 1.0e-5 0.2 0.15

1100 0.2 0.1 0.25 2.2 0.25 500 25 376.991 0.2 0.9e-5 0.25 0.2

Графік(рис. 10. 1):

A graph with green and orange lines

AI-generated content may be incorrect.

Рисунок 10. 1 - Графічне відображення результатів розрахунку

1. **Висновки**

У ході виконання курсової роботи було розроблено та реалізовано програмне забезпечення для моделювання та обчислення потужності N1(t) у системі з урахуванням змінних фізичних параметрів. В основі реалізації лежить математична модель, яка враховує зміну кутової швидкості та висоти рідини в залежності від часу.

Програму реалізовано мовою програмування C++ із використанням базових структур та математичних функцій. Була передбачена можливість зчитування вхідних параметрів з файлу, що забезпечує гнучкість та зручність у використанні. Результати розрахунків виводяться у файл і на екран у зручному форматі з вказанням моменту часу та відповідного значення потужності.

У процесі реалізації враховано критичні ситуації, що можуть виникнути при обчисленнях (зокрема, перевірку на коректність значень, що потрапляють під корінь, а також уникнення ділення на нуль), що свідчить про надійність програми.

Таким чином, поставлені в курсовій роботі завдання було виконано в повному обсязі. Отримані результати можуть бути використані для аналізу й оптимізації технічних систем, що залежать від змінних параметрів у часі.

1. **Список використаної літератури**
2. Відеокурс з програмування [Електронний ресурс] : відеоуроки. – Режим доступу: <https://youtube.com/playlist?list=PL7vq4D0vOpQa9WaLe7btV01eixBUZ6-Ve&si=S8sTSuWjTpQPzOjT>. – Назва з екрана.
3. Методичні вказівки до курсової роботи з дисципліни «Програмування» / уклад.: В. В. Авраменко, В. О. Боровик, Н. В. Тиркусова. – Суми : Сумський державний університет, 2021. – 43 с.
4. Авраменко В. В. Програмування [Електронний ресурс] : навчальний курс / Віктор Васильович Авраменко. – Режим доступу: <https://mix.sumdu.edu.ua/textbooks/104505/index.html>. – Назва з екрана.