МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ



**Дніпропетровський національний університет  
залізничного транспорту імені академіка В. Лазаряна**

Кафедра «Технічна Кібернетика»

**Лабораторна робота №1**

**з дисципліни «Основи программування»**

**на тему: «**Розробка та виконання програми лінійної структури на мові С++. Середовище розробки та виконання програм Microsoft Visual Studio.**»**

Виконав:

студент гр.ПЗ1911

Сафонов Д.Є.

Прийняла: ас. Каф. КІТ

Горбова О.В.

Дніпро, 2019

**Тема.** Розробка та виконання програми лінійної структури на мові С++. Середовище розробки та виконання програм Microsoft Visual Studio.

**Мета.** Ознайомитися з процесом розробки програм. Отримати навички розробки консольного додатка на мові С++ у середовищі Microsoft Visual Studio.

Программа А

1. Постановка завдання

На верхньої основі циліндра з радіусом *R*та висотою *H*розмістили півкулю радіусу *r*, поєднавши вісь симетрії фігур. Обчисліть об’єм та площу поверхні отриманої фігури.

* 1. Зовнішні специфікації
     1. Формат вхідних даних

Вхідні дані: радіус циліндра, радіус сфери і висота циліндра.

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Радіус циліндра | Rc | Дійсне число з області визначення радіуса циліндра | 4 або 2.5 |
| Радіус сфери | Rs | Дійсне число з області визначення радіуса сфери | 4 або 1.5 |
| Висота | h | Дійсне число з області визначення висоти | 5 або 3.6 |

* + 1. Формат вхідних даних

Вихідні дані: об’єм та площа поверхні отриманої фігури.

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| об’єм | v | Дійсне число з області визначення об’єму фігури | 276.32 або 69.865 |
| площа поверхні | s | Дійсне число число з області визначення площі поверхні фігури | 385.17(3) або 96.(5) |

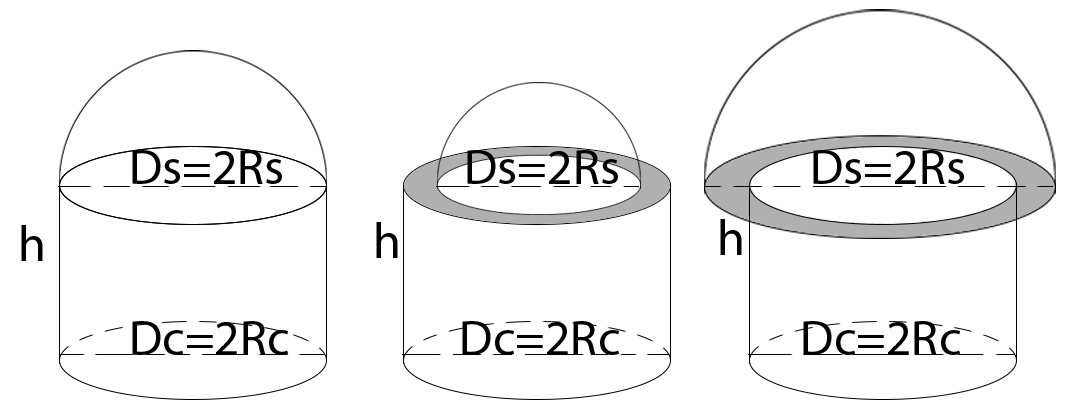
* 1. Функціональні вимоги до програми

Програма повинна реалізовувати такі дії:

– обчислення об’єму та площі поверхні отриманої фігури;

– забезпечення вводу користувачем вхідних даних для обчислення;

– вивід результатів обчислення.

* 1. Вибір методу рішення задачі. 

Rc=Rc – радіус циліндра

Rs=Rs – радіус сфери

H=H – висота циліндра

– площа основи циліндра

– площа основи півсфери

– модуль різниці площі основи циліндра та площі основи півсфери

– довжина окружности циліндра

* 1. Розробка алгоритму програми

а) Словесний алгоритм:

1) Ввести Rc, Rs та h.

5) запис у S0 модуля S0

6) розрахунок за формулами :

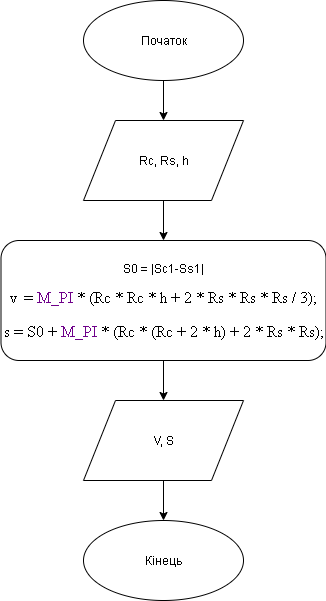
v = M\_PI \* (Rc \* Rc \* h + 2 \* Rs \* Rs \* Rs / 3);

s = S0 + M\_PI \* (Rc \* (Rc + 2 \* h) + 2 \* Rs \* Rs);

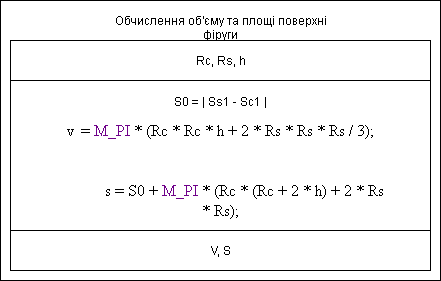
7) вивід результатів

8) Передбачити затримку виконання програми для перегляду результату.

б) Блок-схема:



в) Діаграма N-S:



* 1. Розробка тестів

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Назва | Вхідні дані | | | Очікувані результати | |
|  | Rc | Rs | h | V | S |
| Коректні дані | 4 | 4 | 5 | 385.17(3) | 276.32 |
| Некоректні дані по радіусу циліндра | -1 | 2 | 3 | 26.1(6) | 18.84 |
| Некоректні дані по радіусу сфери | 1 | -2 | 3 | -7.32(6) | 56.52 |
| Некоректні дані по висоті | 1 | 2 | -3 | 7.32(6) | 18.84 |
| Граничний по радіусу циліндра | 0 | 1 | 1 | 2.09(3) | 9.42 |
| Граничний по радіусу сфери | 1 | 0 | 2 | 6.28 | 18.84 |
| Граничний по висоті | 2 | 2 | 0 | 16.74(6) | 37.68 |
| Нульовий | 0 | 0 | 0 | 0 | 0 |

* 1. Текст программи

//Programm calculating Volume of custom shape(Cylinder+Half Sphere)

//Input=Rc,Rs,h(int)

//output=volume,surface area(int)

#include <iostream>

#include <Windows.h>

#ifndef M\_PI

#define M\_PI 3.14 //setting pi value to 3.14

#endif

using namespace std;

//Body

int main()

{

//Rc - Cylinder radius; Rs - sphere radius; S0 - unsigned value of =pi\*(Rc\*Rc-Rs\*Rs); v,s - ans; h- cyl height

cout << "Programm calculating Volume of custom shape(Cylinder+Half Sphere) " << endl;

double Rc, Rs, h;

cout << "input cylinder radius ";

cin >> Rc;

cout << "input sphere radius ";

cin >> Rs;

cout << "input h ";

cin >> h;

double S0;

S0 = M\_PI \* (Rc + Rs) \* (Rc - Rs);

if (S0 < 0) {

S0 = -S0;// making sure that S0>0

}

double v, s;

v = M\_PI \* (Rc \* Rc \* h + 2 \* Rs \* Rs \* Rs / 3);

s = S0 + M\_PI \* (Rc \* (Rc + 2 \* h) + 2 \* Rs \* Rs);

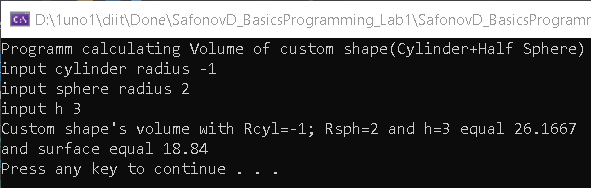
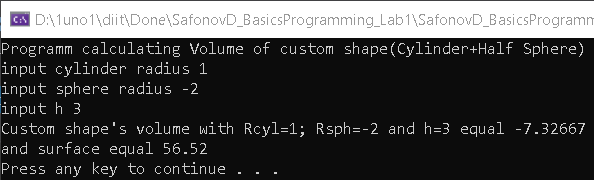
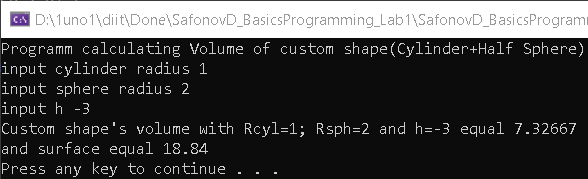
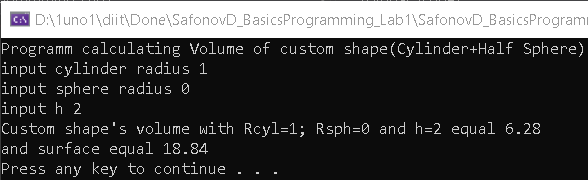
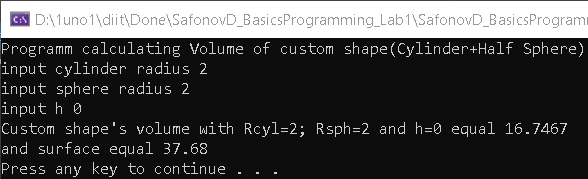
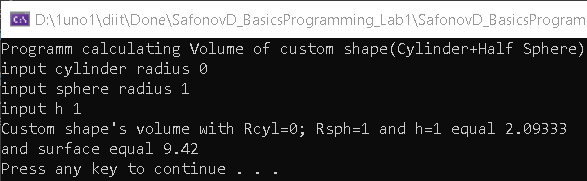
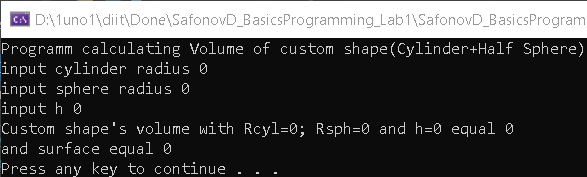
cout << "Custom shape's volume with Rcyl=" << Rc << "; Rsph=" << Rs << " and h=" << h << " equal " << v << endl;

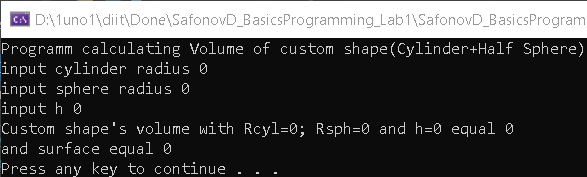
cout << "and surface " "equal " << s << endl;

system("pause");

return 0;

}

1. Результати роботи програми 



Через те що програма не перевіряє дані вводу, вона може давати відповіді, які насправді не існують. Усі відповіді надані програмою співпадають з очікуваними.

Программа B

1. Постановка завдання

Дано чотиризначне число N. Для формування числа M поміняти місцями цифри тисяч і сотень, десятків і одиниць числа N.

По заданому числу Nзнайти число M.

* + 1. Зовнішні специфікації
    2. Формат вхідних даних

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Число N | N | Дійсне чотиризначне число | 1245 |

* + 1. Формат вихідних даних

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Число M | M | Дійсне чотиризначне число | 2154 |

* 1. Функціональні вимоги до програми

Програма повинна реалізовувати такі дії:

– обчислення числа M;

– забезпечення вводу користувачем вхідних даних для обчислення;

– вивід результатів обчислення.

1. Вибір методу рішення задачі.
2. Розробка алгоритму програми

а) Словесний алгоритм:

1) Ввести число N.

2) Обчислити число M за формулами:

d = N % 10;

c = ((N - d) / 10) % 10;

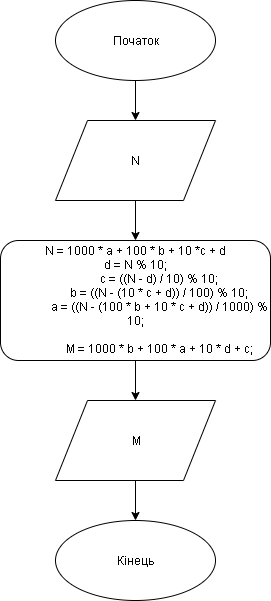
b = ((N - (10 \* c + d)) / 100) % 10;

a = ((N - (100 \* b + 10 \* c + d)) / 1000) % 10;

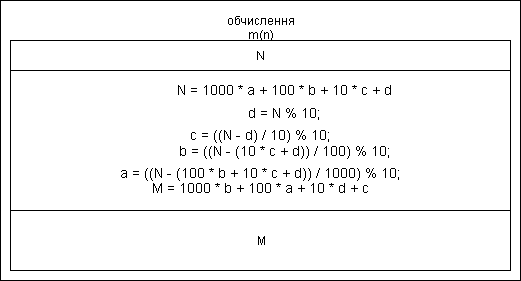
3) Вивести результат M на екран.

4) Передбачити затримку виконання програми для перегляду результату.

б) Блок-схема:



в) Діаграма N-S:



1. Розробка тестів

|  |  |  |
| --- | --- | --- |
| Назва | Вхідні дані | Очікувані результати |
|  | N | M |
| Коректні дані | 1245 | 2154 |
| Тризначне число | 689 | 6098 |
| П’ятизначне число | 26784 | 7648 |

1. Текст програми

//programm making M=badc from N=abcd

#include <iostream>

#include <Windows.h>

using namespace std;

int main()

{

long int N, M, a, b, c, d;

cout << "programm making M=badc from N=abcd" << endl << "input N ";

cin >> N;

while (N != 0)//loop to split number into digits

{

d = N % 10;

c = ((N - d) / 10) % 10;

b = ((N - (10 \* c + d)) / 100) % 10;

a = ((N - (100 \* b + 10 \* c + d)) / 1000) % 10;

M = 1000 \* b + 100 \* a + 10 \* d + c;

cout << "M from given N is equal " << M << endl;

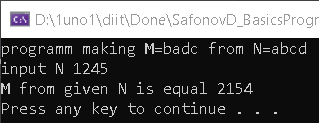
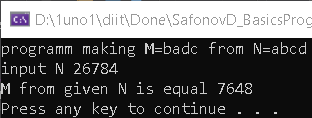
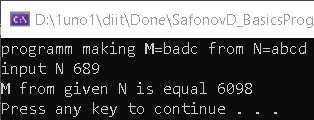
system("pause");

return 0;

}

}

1. Результати роботи програми



Відповіді надані програмою співпадають з очікуваними.

Программа C

* 1. Постановка завдання

*-64a3 + 240a2b - 300ab2+ 125b3*

Обчислити вираз для заданих значень *a*і *b*.

1. Зовнішні специфікації
   * 1. Формат вхідних даних

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Число a | a | Дійсне число з області визначення a | 5 |
| Число b | b | Дійсне число з області визначення b | 6 |

* + 1. Формат вихідних даних

|  |  |  |  |
| --- | --- | --- | --- |
| Найменування даних | Умовне позначення | Вимоги до данних | Приклад |
| Результат розрахунків | r | Дійсне число з області визначення r | 1000 |

* + 1. Функціональні вимоги до програми

Програма повинна реалізовувати такі дії:

– обчислення результату(число r);

– забезпечення вводу користувачем вхідних даних для обчислення;

– вивід результатів обчислення.

1. Вибір методу рішення задачі.
   1. Розробка алгоритму програми

а) Словесний алгоритм:

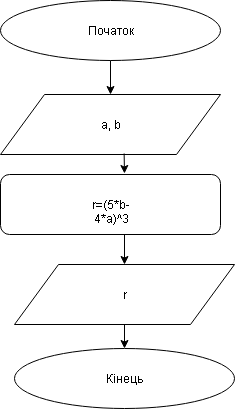
1) Ввести числа a, b.

2) Обчислити число r за формулою r = (5 \* b – 4 \* a)^3.

3) Вивести результат p на екран.

4) Передбачити затримку виконання програми для перегляду результату.

б) Блок-схема:



в) Діаграма N-S:

|  |  |  |  |
| --- | --- | --- | --- |
| Обчислення числа r   |  | | --- | | Ввод a, b | | r=(5b-4a)^3 | | Вивід r | |

1. Розробка тестів

|  |  |  |  |
| --- | --- | --- | --- |
| Назва | Вхідні дані | | Очікувані результати |
|  | a | b | r |
| Коректні дані | 5 | 6 | 1000 |
| Нульові | 0 | 0 | 0 |

1. Текст програми

//programm calculating -64a^3 + 240a^(2)\*b - 300ab^2 + 125b^3

#include <Windows.h>

#include <iostream>

using namespace std;

int main()

{

float a, b;

cout << "programm calculating - 64a ^ 3 + 240a ^ (2) \* b - 300ab ^ 2 + 125b ^ 3 from given a and b variables" << endl;

cout << "input a ";

cin >> a;

cout << "input b ";

cin >> b;

float r;

r = (5 \* b - 4 \* a) \* (5 \* b - 4 \* a) \* (5 \* b - 4 \* a);

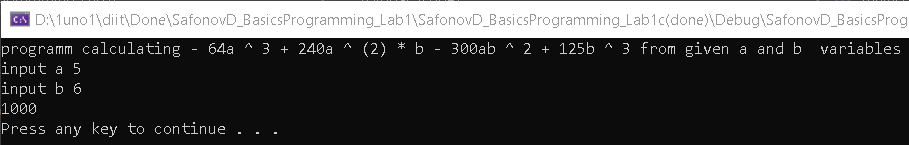
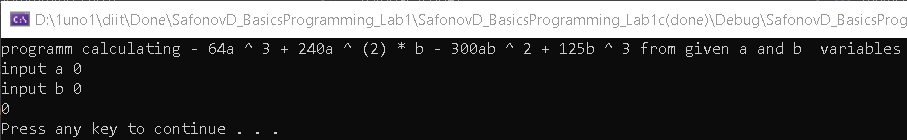
cout << r << endl;

system("pause");

return 0;

}

1. Результати роботи програми



Відповіді надані програмою співпадають з очікуваними.

Висновок

Для виконання даної лабораторної роботи я вивчив синтаксис мови програмування C ++, її базові команди введення-виведення і математичних розрахунків, опису змінних; Навчився тестувати свої програми на коректність виконання.

Для розробки програм довелося використати математичні формули, а також вивести з них формули, які працюють для моїх конкретних завдань, у деяких місцях були оголошенні необов’язкові змінні, для полегшення читання коду.

В останній програмі заданий приклад був спрощений для оптимізації розрахунків, завдяки цьому код став меншим и виконується швидше.