**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ**

**Чорноморський національний університет   
імені Петра Могили**

**Факультет комп’ютерних наук**

**Кафедра інженерії програмного забезпечення**

**ЗВІТ**

*з лабораторної роботи № 4*

**«Програмування лінійних алгоритмів. Приведення типів даних** **»**

**Завдання № 7**

Дисципліна «Основи програмування»

Спеціальність: **Інженерія програмного забезпечення**

121-ЛР.ПЗ.04-109.12314286

***Cтудентка***

*\_\_\_\_\_\_\_\_\_\_\_\_\_Н.Ю.Воронцова*

*(підпис)*

*\_\_01.10.2018 \_\_*

*(дата)*

***Викладач*** *\_\_\_\_\_\_\_\_\_\_\_\_\_Є. О. Давиденко*

*(підпис)*

*\_\_\_\_\_\_\_\_\_\_\_\_\_*

*(дата)*

Миколаїв – 2018

**Лабораторна робота №4**

|  |  |
| --- | --- |
| *Тема роботи* – | Програмування алгоритмів, що розгалужуються |

|  |  |
| --- | --- |
| *Мета роботи* – | Ознайомитись з програмуванням алгоритмів, що розгалужуються. Засвоїти методи їх використання. |

Завдання 1:

Виконати завдання 2 до лр. №3 з урахуванням ОДЗ.



**

**Алгоритмізація завдання 1:**

Блок-схема алгоритму обчислення виразу a з урахуванням області допустимих значень.

Початок

Введення x, z, y.



Так Ні

Кінець

Виведення a.

Виведення «a неможливо обрахувати».

Рисунок 1 – блок-схема алгоритму обчислення виразу а з урахуванням ОДЗ

Лістинг 1 завдання 1:

#include "pch.h"

#include "windows.h"

#include "math.h"

#include "iostream"

#define \_USE\_MATH\_DEFINES

using namespace std;

int main()

{

double a, x, y, z;

cout << "Input x\n";

cin >> x;

cout << "Input y\n";

cin >> y;

cout << "Input z\n";

cin >> z;

if ((z + exp(2 + x)) >= 0)

{

a = pow(x, 2)\*(x + 1) / pow((z + exp(2 + x)), 1. / 3.) - pow(sin(x + y), 2);

cout << "Result = " << a;

}

system("pause");

return 0;

}

**Результат роботи програми:**

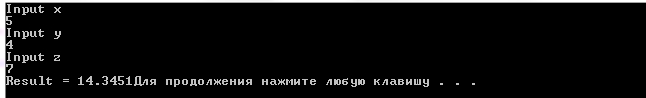


Рисунок 2 – блок-схема алгоритму обчислення виразу a з урахуванням ОДЗ

Блок-схема алгоритму обчислення виразів b з урахуванням області допустимих значень.

Початок

Введення x, z, y.

**

Z >0

Так Ні

Кінець

Виведення b.

Виведення «b неможливо обрахувати».

Рисунок 3 – блок-схема алгоритму обчислення виразу b з урахуванням ОДЗ

Лістинг 2 завдання 1:

#include "pch.h"

#include "windows.h"

#include "math.h"

#define \_USE\_MATH\_DEFINES

#include <iostream>

using namespace std;

int main()

{

setlocale(0, "");

double x, y, z, a, b;

printf("Input x\n x=");

scanf\_s("%lf", &x);

printf("Input y\n y=");

scanf\_s("%lf", &y);

printf("Input z\n z=");

scanf\_s("%lf", &z);

b = sqrt(abs(x - 6 \* y) / z) + pow((cos(x + z), 3), 2) + sin(M\_PI \* 45 / 180);

if (z <= 0) printf("b невозможно посчитать. \n");

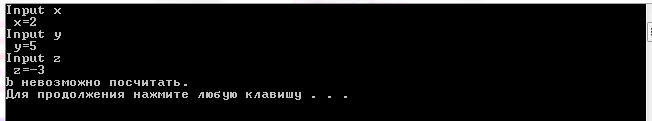
else printf("b=%lf\n", b);

system("pause");

return 0;

}

**Результат роботи програми:**



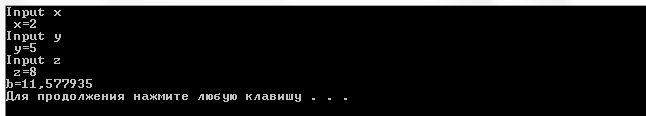


Рисунок 4 – Тестування лістингу 2 до завдання 1

**Завдання 2.**

Використовуючи оператор **if** знайдіть значення виразу за варіантом.

Розрахувати значення функції F в залежності від значень (a, b, c, x), де (a, b, c, x) дійсні числа.

Реалізувати два варіанти програмного коду:

а) розрахунок функції F без застосування логічних операцій;

б) розрахунок функції F із застосування логічних операцій.



**Алгоритмізація завдання 2:**

Початок

Введення a, x, b, c, D E, N .

Кінець

X<0, b≠0

-c ≠ 0

E=

Виведення E.

N= 

Виведення «Error».

Виведення N.

\_ +

D = 

\_ +



Виведення D.

\_

+

Рисунок 5 – блок-схема алгоритму знаходження

Лістинг 1 завдання 2: (з використанням логічних операцій):

#include "pch.h"

#include "windows.h"

#include "math.h"

#include "iostream"

#define \_USE\_MATH\_DEFINES

using namespace std;

int main()

{

double x, a, b, c, D, E, N;

printf("Input x\n x=");

scanf\_s("%lf", &x);

printf("Input a\n a=");

scanf\_s("%lf", &a);

printf("Input b\n b=");

scanf\_s("%lf", &b);

printf("Input c\n c=");

scanf\_s("%lf", &c);

if ((x < 0) && (b != 0)) {

D = -a \* pow(x, 2) + b;

printf("D=%f\n", D);

}

else if ((x > 0) && (b = 0)) {

E = x / (x - c) + 5.5;

printf("E=%f\n", E);

}

else if (-c != 0)

{

N = x / -c;

printf("N=%f\n", N);

}

else

{

printf("Error\n");

}

system("pause");

return 0;

}

**Результат роботи програми:**

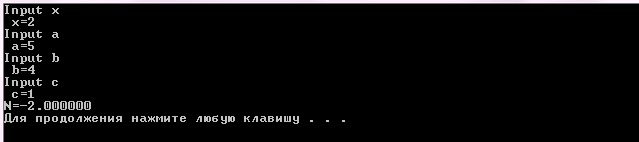


Рисунок 6 – Тестування лістингу 1 до завдання 2

Лістинг 2 завдання 2: (без використання логічних операцій):

#include "pch.h"

#include "windows.h"

#include "math.h"

#include "iostream"

#define \_USE\_MATH\_DEFINES

using namespace std;

int main()

{

double x, a, b, c, D, E, N;

printf("Input x\n x=");

scanf\_s("%lf", &x);

printf("Input a\n a=");

scanf\_s("%lf", &a);

printf("Input b\n b=");

scanf\_s("%lf", &b);

printf("Input c\n c=");

scanf\_s("%lf", &c);

if (x < 0)

if (b != 0) {

D = -a \* pow(x, 2) + b;

printf("D=%f\n", D);

}

else if (x > 0)

if (b = 0) {

E = x / (x - c) + 5.5;

printf("E=%f\n", E);

}

else if (-c != 0)

{

N = x / -c;

printf("N=%f\n", N);

}

else

{

printf("Error\n");

}

system("pause");

return 0;

}

**Результат роботи програми:**



Рисунок 7 – Тестування лістингу 2 до завдання 2

**Завдання 3**

Використовуючи оператор **switch** знайдіть значення виразу за варіантом.

Написати алгоритм, що дозволяє отримати словесне найменування шкільних оцінок.

Початок

Введення n.

n

1

Виведення «один»

2

Виведення «два»

3

Виведення «три»

Виведення «чотири»

4

Виведення «п’ять»

5

6

Виведення «шість»

7

Виведення «сім»

8

Виведення «вісім»

Виведення «дев’ять»

9

Виведення «десять»

10

Виведення «одинадцять»

11

Виведення «дванадцять»

12

Інше число

Виведення «Error»

Кінець

Рисунок 8 – блок-схема алгоритму словесного найменування шкільних оцінок.

Лістинг завдання 3:

#include "pch.h"

#include "windows.h"

#include <iostream>

using namespace std;

int main()

{

setlocale(0, "");

int n;

cout << "Введите нужную оценку: ";

cin >> n;

switch (n)

{

case 1:cout << "Один\n"; break;

case 2:cout << "Два\n"; break;

case 3:cout << "Три\n"; break;

case 4:cout << "Четыре\n"; break;

case 5:cout << "Пять\n"; break;

case 6:cout << "Шесть\n"; break;

case 7:cout << "Семь\n"; break;

case 8:cout << "Восемь\n"; break;

case 9:cout << "Девять\n"; break;

case 10:cout << "Десять\n"; break;

case 11:cout << "Одинадцать\n"; break;

case 12:cout << "Двенадцать\n"; break;

default:cout << "Числу, которое вы ввели не соответствуют никакая оценка\n";

}

system("pause");

return 0;

}

**Результат роботи програми:**

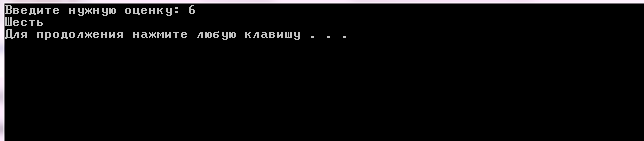


Рисунок 9 – Тестування лістингу до завдання 3.

**Висновок**: під час виконання лабораторної роботи №4 було розглянуто методи програмуванням алгоритмів, що розгалужуються. У роботі для виконання завдання було використано оператори if else та switch. В результаті була створена програма для обчислення виразів з урахуванням їх області допустимих значень та програма, що дозволяє отримати словесне найменування шкільних оцінок.