**ЛАБОРАТОРНА РОБОТА № 4**

**РОЗРОБКА ПРОГРАМ З ЦИКЛІЧНИМИ АЛГОРИТМАМИ**

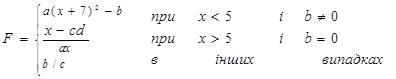
**Мета роботи**

* Оволодіти навичками щодо використання різних видів операторів циклу: цикли з параметрами *for*, цикли з умовою *while*, цикли з післяумовою *do*-*while*.
* Дослідити роботу операторів керування: *break, continue, return, goto.*

**Варіант 14**

**Задача 1**

Обчислити і вивести на екран у вигляді таблиці значення функції F на інтервалі від Хпоч. до Хкінц. з кроком Δх. Значення a, b, c, Хпоч., Хкінц., Δх вводити з клавіатури з обов’язковою перевіркою правильності введення.



**Формалізація задачі**

1. Вхідні дані: X0, Xk, x, y, h, d, a, b, c.
2. Вихідні дані: F
3. Типи даних. Оскільки значення змінних можуть бути дійсними, то тип даних double.
4. Перевірка правильності. Для перевірки правильності обчислень, слід підготувати декілька контрольних прикладів.

Кінець

F = xp / c - c / xp;

F = (xp - a) / (xp - c);

**Лістинг програми:**

#include <iostream>

#include <iomanip>

#include <cmath>

using namespace std;

int main() {

int t;

float X0, Xk, x, y, h, d, a, b, c;

cout << "Enter data:" << endl;

cout << "t = ";

cin >> t;

cout << "a = ";

cin >> a;

cout << "b = ";

cin >> b;

cout << "c = ";

cin >> c;

cout << "d = ";

cin >> d;

cout << "Enter Xnach, Xend, h: ";

cin >> X0 >> Xk >> h;

if (h == 0) {

cout << "Error: Step size (h) cannot be zero." << endl;

return 1;

}

cout << "\nInterval [" << X0 << ", " << Xk << "]\nMove h = " << h << endl;

cout << "\n----------------------\n| x | y |" << endl;

while (X0 < Xk) {

x = X0;

if (x < 5) {

if (a == 0) {

cout << "Error: Division by zero (a \* x when x < 5)." << endl;

return 1;

}

y = a \* pow((x + 7), 2) - b;

}

else if (x > 5) {

if (a \* x == 0) {

cout << "Error: Division by zero (a \* x when x > 5)." << endl;

return 1;

}

y = (x - c \* d) / (a \* x);

}

else {

if (c == 0) {

cout << "Error: Division by zero (c when x = 5)." << endl;

return 1;

}

y = b / c;

}

cout << "| " << fixed << setprecision(2) << setw(8) << x << " | " << setw(8) << y << " |" << endl;

X0 += h;

}

return 0;

}

**Результати:**

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

**Задача 2**

Використовуючи оператори циклу, обчислити значення виразів, якщо *х –*дійсне, *n* – натуральне число. Числа *х*і *n* вводяться з клавіатури.





**Формалізація задачі**

1. Вхідні дані: a,b,x, n.
2. Вихідні дані: a,b;
3. Типи даних. Оскільки значення змінних можуть бути дійсними, то тип даних double.
4. Перевірка правильності. Для перевірки правильності обчислень, слід підготувати декілька контрольних прикладів.

**Лістинг програми:**

#include <iostream>

#include <cmath>

using namespace std;

int main() {

double a = 0, b = 1, n, x;

setlocale(LC\_ALL, "ukr");

do {

cout << "Введiть натуральне число: ";

cin >> n;

if (n <= 0)

{

cout << "Це не натуральне число!" << endl;

}

} while (n <= 0);

cout << "Введiть x=";

cin >> x;

for (int k = 1; k <= n; k++) {

a += pow(k, k) \* pow(x, (2 \* k));

}

cout << "Сума " << a << endl;

for (int i = 1; i <= 10; i++) {

int fact\_i;

for (int fi = 1; fi <= i; fi++)

{

fact\_i \*= fi;

}

if (fact\_i == 0) {

cout << "Помилка: ділення на нуль (факторіал 0)" << endl;

return 1;

}

double dil = pow(i, 2) + 2 \* i + 3;

if (dil == 0) {

cout << "Помилка: ділення на нуль (знаменник 0)" << endl;

return 1;

}

b \*= fact\_i / dil;

}

cout << "Добуток " << b << endl;

return 0;

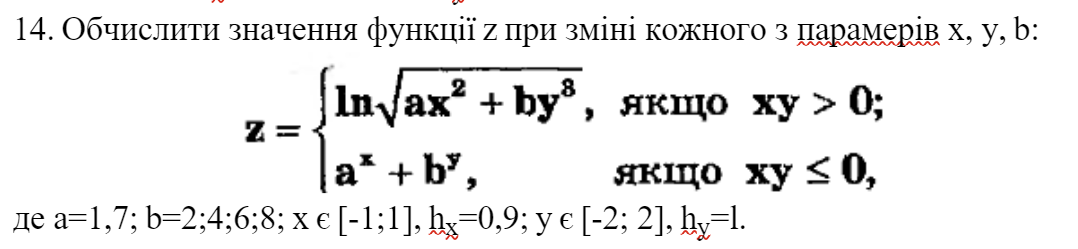
}

**Результати:**

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

**Задача 3**

Обчислити функцію у та знайти її найбільше значення:

****

якщо b=1,2; а є [1;2], ha=0,5; х є [0;1], hx = 0,2.

**Формалізація задачі**

1. Вхідні дані: відсутні
2. Вихідні дані: max
3. Типи даних. Оскільки значення змінних можуть бути дійсними, то тип даних double.
4. Перевірка правильності. Для перевірки правильності обчислень, слід підготувати декілька контрольних прикладів.

**Блок-схема алгоритму:**

**Лістинг програми:**

#include <iostream>

using namespace std;

int main() {

double f, a = 1.7, bp = 2, bk = 8, xp = -1, xk = 1, hx = 0.9, yp = -2, yk = 2, hy = 1;

for (double i = xp; i <= xk; i += hx) {

for (double j = yp; j <= yk; j += hy) {

cout << endl;

for (double b = bp; b <= bk; b += 2) {

if (i \* j > 0) {

f = log(sqrt(a \* pow(i, 2) + b \* pow(j, 3)));

if (sqrt(a \* pow(i, 2) + b \* pow(j, 3)) >= 0) {

cout << "x = " << i << "\ty = " << j << "\tb = " << b << "\tF = " << f << endl;

}

}

else {

f = pow(a, i) + pow(b, j);

cout << "x = " << i << "\ty = " << j << "\tb = " << b << "\tF = " << f << endl;

}

}

}

}

}

**Результати:**



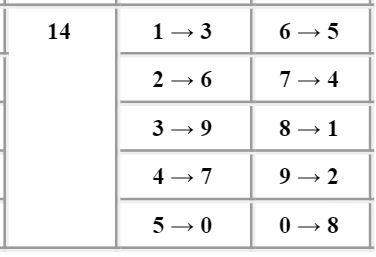
**Задача 4**

Один з найпотужніших методів захисту інформації – це її шифру­вання. Розробити програму, яка для цілих D:\Технологія програмування2_files\image111.png, що вводяться з клавіатури у зашифрованому вигляді, обчислюватиме вираз

D:\Технологія програмування2_files\image112.png**, (** де mod 10 – остача від ділення на 10**)**

**та шифруватиме результат перед виведенням на екран**

**(**шифрування та розшифрування реалізувати у підпрограмах функціях**):**



**Формалізація задачі**

1. Вхідні дані: b, c
2. Вихідні дані: a
3. Типи даних. Оскільки значення змінних можуть бути цілими, то int.
4. Перевірка правильності. Для перевірки правильності обчислень, слід підготувати декілька контрольних прикладів.

**Лістинг програми:**

#include <iostream>

using namespace std;

int deshifrator(int a)

{

switch (a)

{

case 0:a = 5; break;

case 1:a = 8; break;

case 2:a = 9; break;

case 3:a = 1; break;

case 4:a = 7; break;

case 5:a = 6; break;

case 6:a = 2; break;

case 7:a = 4; break;

case 8:a = 0; break;

case 9:a = 3; break;

}

return a;

}

int shifrator(int a)

{

a = abs(a);

switch (a)

{

case 0:a = 8; break;

case 1:a = 3; break;

case 2:a = 6; break;

case 3:a = 9; break;

case 4:a = 7; break;

case 5:a = 0; break;

case 6:a = 5; break;

case 7:a = 4; break;

case 8:a = 1; break;

case 9:a = 2; break;

}

return a;

}

int main()

{

setlocale(LC\_ALL, "ukr");

int b, c, S = 0, a;

bool ind;

do

{

ind = true;

cout << "Введiть значення b=";

cin >> b;

cout << "Введiть значення c=";

cin >> c;

if (b < 0 || c < 0 || b>9 || c>9)

{

cout << "Введенi недопустимi значення змiнних!" << endl;

ind = false;

}

} while (ind == false);

b = deshifrator(b);

c = deshifrator(c);

for (int i = 1; i <= 10; i++)

{

S += b \* i + 2 \* pow(c - i, i);

}

a = S % 10;

a = shifrator(a);

cout << "a=" << a << endl;

}

**Результати:**

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

**Висновок**: Я оволодів навичками щодо використання різних видів операторів циклу: цикли з параметрами *for*, цикли з умовою *while*, цикли з післяумовою *do*-*while* і також дослідив роботу операторів керування: *break, continue, return, goto.*