ЗВІТ

З ЛАБОРАТОРНОЇ РОБОТИ

за курсом «Програмування»

студентки групи ПС-21-1

Авраменко Анни Сергіївни

кафедра комп’ютерних технологій, ДНУ

2020/2021

• Постановка задачі

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generated

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, email

Description automatically generatedText

Description automatically generatedGraphical user interface, text, application, email

Description automatically generated

• Опис розв’язку

В задачах SUM, COUNTODD, PRODUCT, MAX застосовується цикл While(true) та оператор Break, що завершує виконання циклу, коли вводиться 0 з клавіатури. Використовувати масив для таких задач є недоцільним, оскільки нам не потрібно пам’ятати усі числа, а досить лише попереднього та поточного, що вміщаються у 2 змінні відповідних типів.

Задачі Fact, Fact2 та Fibbonaci використовую рекурентну функцію та декілька перевірок на правильний ввід даних, що вводить користувач. Це також можна було зробити циклом та без методу, оскільки в програмі ця функція використовується лише 1 раз, але мною був вибраний цей метод розв’язку, через його зручність та компактність.

Задача HIT2 використовує математичну формулу кола, та перевіряє, чи є сума квадратів різниці координат х та у попарно більше або менше квадрата радіусу кола. Також, центр кола генерується випадково.

• Вихідний текст програми розв’язку задачі(основні фрагменти з коментарями)

#include <iostream>

using namespace std;

int main()

{

int sum = 0, counter = 0;

while (1)

{

int a;

cout << "Enter the number = ";

cin >> a;

if (a != 0) {

a % 2 == 1 ? counter++ : true;

}

else

break;

}

switch (counter) {

case 0:

cout << "There were no odd numbers";

break;

case 1:

cout << "There was " << counter << " odd number";

break;

default:

cout << "There were " << counter << " odd numbers";

}

}

#include <iostream>

long int Fact(int n) {

return n <= 1 ? 1 : n \* Fact(n - 1);

}

int main()

{

int n;

std::cin >> n;

std::cout << Fact(n);

}

#include <iostream>

#include <math.h>

int Fact(int n) {

return n <= 1 ? 1 : n \* Fact(n - 1);

}

using namespace std;

int main()

{

double a;

double n;

cout << "N = ";

cin >> n;

if (!cin) {

cout <<"Error: N must a number, no a string!";

return 0;

}

else if (n < 0) {

cout << "Error: N must be positive number!";

return 0;

}

else if (modf(n, &a) != 0.0) {

cout << "Error: N must be natutral number!";

return 0;

}

cout << n << "! = " << Fact(n);

}

// fibonacci.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

using namespace std;

int fibonacci(int x) {

int res;

((x == 1) || (x == 0)) ?res = x : res = (fibonacci(x - 1) + fibonacci(x - 2));

return res;

}

int main()

{

int n;

cout << "N = ";

cin >> n;

cout << "F(" << n << ") = " << fibonacci(n);

}

// hit2.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

#include <stdlib.h>

#include <time.h>

using namespace std;

int main()

{

double x0, y0, r, x = 0, y = 0;

double temp = 0;

srand(time(NULL));

x0 = (rand() % 100 - 60) / 10.0;

y0 = (rand() % 100 - 60) / 10.0;

r = (rand() % 50) / 10.0;

cin >> x >> y;

while ((x - x0)\*(x - x0) + (y - y0)\*(y - y0) > r\*r) {

cout << "Try again! ";

temp = (x - x0)\*(x - x0) + (y - y0)\*(y - y0);

cin >> x >> y;

if (temp > (x - x0)\*(x - x0) + (y - y0)\*(y - y0))

cout << "Closer!\n";

else

cout << "Farther!\n";

}

cout << "Point (" << x << ", " << y << ") is in circle centered at a point (" << x0 << ", " << y0 << ") radius " << r << "\n";

}

// Sum.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

using namespace std;

int main()

{

int a, counter = 0;

int temp =-2147483647;

while (1)

{

counter++;

cout << "a["<<counter<<"] = ";

cin >> a;

if (a != 0) {

a > temp ? temp = a : true;

}

else

break;

}

cout << "MAX=" << temp;

}

#include <iostream>

using namespace std;

int main()

{

setlocale(LC\_CTYPE, "ru");

int selectedMenu = -1;

while (selectedMenu != 0)

{

cout << "The information system of the Faculty of Applied Mathematics is working!\n";

cout << "1. News of the day\n";

cout << "2. Аnecdote of the week\n";

cout << "3. Author!!!\n";

cin >> selectedMenu;

switch (selectedMenu)

{

case 1:

cout << "1. News of the day\n";

break;

case 2:

cout << "2. Аnecdote of the week\n ";

cout << "Аnecdote!\n ";

break;

case 3:

cout << "3. Author!!!\n";

break;

case 0:

break;

default:

break;

}

}

}

// parrot2.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

using namespace std;

int main()

{

char st[256]="";

while (true)

{

cin.getline(st,256);

if (st[0] == '\0')

break;

cout << st;

}

}

// Sum.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

using namespace std;

int main()

{

int prod = 1, counter = 0;

while (1)

{

int a;

counter++;

cout << "a[" << counter << "] = ";

cin >> a;

if (a != 0)

prod \*= a;

else

break;

}

cout << "PRODUCT=" << prod;

}

// Sum.cpp : Этот файл содержит функцию "main". Здесь начинается и заканчивается выполнение программы.

//

#include <iostream>

using namespace std;

int main()

{

int sum = 0, counter = 0;

while (1)

{

int a ;

counter++;

cout << "a[" << counter << "] = ";

cin >> a;

if (a != 0)

sum += a;

else

break;

}

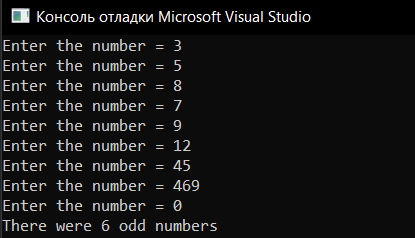
cout <<"SUM="<< sum;

}

• Опис інтерфейсу програми(керівництво користувача)

Інтерфейс програми складається з консольних додатків, в якому виводиться результат програми згідно до даних, що вводить користувач.

• Опис тестових прикладів

Graphical user interface, text

Description automatically generatedText

Description automatically generated with low confidence

Text, website

Description automatically generated

Graphical user interface, website

Description automatically generated

A screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidence

Text

Description automatically generated

Text

Description automatically generated

Text

Description automatically generatedA screenshot of a computer

Description automatically generated with medium confidence

Text

Description automatically generated with medium confidence

Graphical user interface, text

Description automatically generatedText

Description automatically generated