Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерного проектирования

Кафедра Информатики

Дисциплина «Программирование»

**ОТЧЕТ**

к лабораторной работе №2

на тему:

**«Управление потоком выполнения программы»**

БГУИР 6-05-0612-02 36

|  |
| --- |
| Выполнил студент группы 353505  ДАНИЛОВ Дмитрий Игоревич |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

Задание 1. Вариант 8.Написать программу, которая определяет является ли сумма цифр двухзначного числа четной.

Задание 2. Вариант 8.Дана точка на плоскости с координатами (х, у). Составить программу, которая выдает одно из сообщений "Да", "Нет", "На границе" в зависимости от того, лежит ли точка внутри заштрихованной области, вне заштрихованной области или на ее границе. Область задаётся графически (рисунок 1).

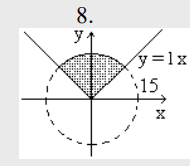


Рисунок 1 – Условие задания 2

# 2 Выполнение работы

Для выполнения работы был создан проект, в решение которого находится 2 задачи. Для обоих заданий в первую очередь надо было предусмотреть то, чтобы программа имела возможность продолжить свое выполнение и важно, чтобы это было реализовано через цикл и условный оператор switch. Этот пункт заданий был реализован при помощи оператора цикла while, в качестве аргумента значение true, но для того чтобы цикл не выполнялся вечно нужно было добавить условие выхода из цикла, это было реализовано через условный оператор switch, в качестве аргумента принимает строку, которая запрашивается на ввод в консоль (рисунок 2). Если пользователь хочет ввести данные повторно (рисунок 3), если не хочет (рисунок 4). Также, если пользователь ввел некорректное число (рисунок 5). Далее для обоих заданий была написана основная логика программы.

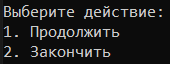


Рисунок 2 – Запрос на повторный ввод данных

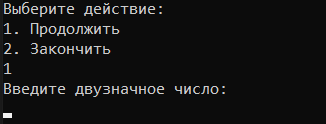


Рисунок 3 – Результат работы программы при желании пользователя ввести данные повторно

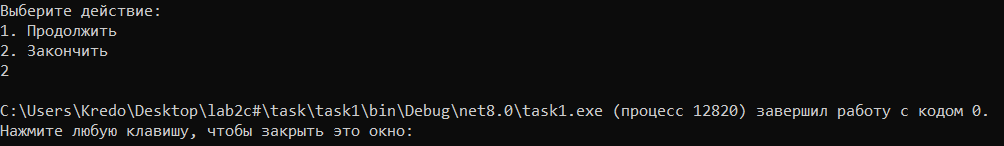


Рисунок 4 – Результат работы программы при нежелании пользователя ввести данные повторно



Рисунок 5 – Результат работы программы при вводе пользователем некорректных данных

Задание 1. Вариант 8.

while (true)

{

Console.WriteLine("Введите двузначное число:");

string input = Console.ReadLine();

if (!int.TryParse(input, out int number) || Math.Abs(number) < 10 || Math.Abs(number) > 99)

{

Console.WriteLine("Некорректный ввод. Пожалуйста, введите двузначное число.");

continue;

}

int digit1 = Math.Abs(number) / 10;

int digit2 = Math.Abs(number) % 10;

int sum = digit1 + digit2;

Console.WriteLine($"Сумма цифр: {sum}");

switch (sum % 2)

{

case 0:

Console.WriteLine("Сумма цифр является четной.");

break;

case 1:

Console.WriteLine("Сумма цифр является нечетной.");

break;

}

Console.WriteLine("Выберите действие:");

Console.WriteLine("1. Продолжить");

Console.WriteLine("2. Закончить");

string choice = Console.ReadLine();

switch (choice)

{

case "1":

continue;

case "2":

return;

default:

Console.WriteLine("Некорректный выбор. Пожалуйста, попробуйте снова.");

break;

}

}

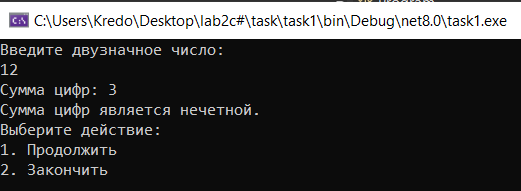


Рисунок 6 – Результат работы программы 1 при вводе числа, у которого сумма цифр нечетная

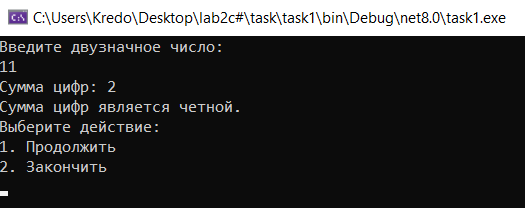


Рисунок 7 – Результат работы программы 1 при вводе числа, у которого сумма цифр четная

Задание 2. Вариант 8.

while (true)

{

Console.WriteLine("Введите значение x: ");

string input = Console.ReadLine();

bool result = double.TryParse(input, out var x);

if (!result)

{

Console.WriteLine("Ошибка");

continue;

}

Console.WriteLine("Введите значение y: ");

input = Console.ReadLine();

result = double.TryParse(input, out var y);

if (!result)

{

Console.WriteLine("Ошибка");

continue;

}

double r = Math.Sqrt(x \* x + y \* y);

double theta = Math.Atan2(y, x);

if (r > 15 || theta < Math.PI / 4 || theta > 3 \* Math.PI / 4)

{

Console.WriteLine("Нет");

}

else if (r == 15 || theta == Math.PI / 4 || theta == 3 \* Math.PI / 4)

{

Console.WriteLine("На границе");

}

else

{

Console.WriteLine("Да");

}

Console.WriteLine("Выберите действие:");

Console.WriteLine("1. Продолжить");

Console.WriteLine("2. Закончить");

string choice = Console.ReadLine();

switch (choice)

{

case "1":

continue;

case "2":

return;

default:

Console.WriteLine("Некорректный выбор. Пожалуйста, попробуйте снова.");

break;

}

}

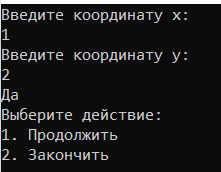


Рисунок 9 – Результат работы программы 2 при вводе координат точки,

лежащей в выделенной области

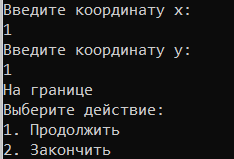


Рисунок 10 – Результат работы программы 2 при вводе координат точки, лежащей на границе выделенной области

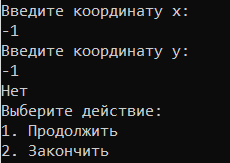


Рисунок 11 – Результат работы программы 2 при вводе координат точки, не лежащей в выделенной области

# Вывод

В ходе лабораторной работы были изучены способы управления потоком выполнения программы такие, как логические операторы и операторы циклов.