Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерных систем и сетей

Кафедра Информатики

Дисциплина «Программирование»

**ОТЧЕТ**

к лабораторной работе №3

на тему:

**«Классы математических функций, классы для работы с числами и датами»**

БГУИР 6-05-0612-02 113

|  |
| --- |
| Выполнил студент группы 453503  ХАЛАМОВ Николай Андреевич |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2025

# 1 Индивидуальное задание

**Задание 1. Вариант 3.** Описать класс (с произвольным именем) в отдельном файле проекта. В созданном классе: разработать метод f(x), который вычисляет значение по следующей формуле:f(x)=x-sinx. Определить, в какой из точек а или b, функция принимает наименьшее значение.

**Задание 2. Вариант 1.** Добавить в проект папку Services. В созданной папке описать класс (с произвольным именем). В созданном классе описать метод, вычисляющий значение функции в соответствии с индивидуальным вариантом. Метод должен вывести в консоль результат выполнения и номер ветки, по которой производилось вычисление (см. рисунок 1).

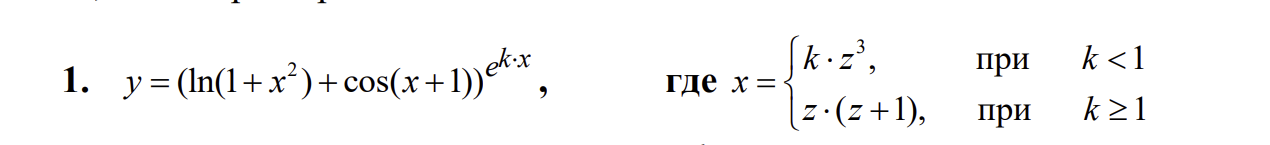


Рисунок 1 – Индивидуальное задание

**Задание 3.** В отдельном файле описать класс DateService. Класс должен предоставлять следующие методы: Определять день недели произвольной даты, передаваемой методу в качестве строкового параметра (например GetDay(string date) Определять, сколько дней пройдет между текущей датой и датой, передаваемой методу в качестве числовых параметров: день, месяц, год (например GetDaysSpan(int day, int month, int year).

# 2 Выполнение работы

Для первого задания были созданы 5 классов для вывода консольного меню с информацией о задании, для ввода чисел и проверки корректности ввода, для метода, вычисляющего выражение по условию, для всех функции в общем, для возможности повторного ввода значений в консоль. Был использован метод класса Math.Sin для вычисления синуса.

public static double F(double x)

{

double value = Math.Sin(x) - x;

return value;

}

Был использован оператор if else для повторного введения значений в консоль. В классе Task реализована проверка того, в какой из точек значение функции больше.

using System;

namespace \_453503\_Халамов

{

internal class Task

{

public static void MainTask()

{

Display.ConsoleMenu();

double a = Input.GetNumber("a");

double b = Input.GetNumber("b");

double fa = Function.F(a);

double fb = Function.F(b);

Console.WriteLine($"Значение функции в точке a равно {fa}");

Console.WriteLine($"Значение функции в точке b равно {fb}\n");

if (fa > fb)

{

System.Console.WriteLine("Наименьшее значение функция принимает в точке b");

}

else if (fa < fb)

{

System.Console.WriteLine("Наименьшее значение функция принимает в точке a");

}

else if (fa == fb)

{

System.Console.WriteLine("Значения f(a) и f(b) равны");

}

}

}

}

Был продемонстрирован результат работы программы (см. рисунок 2).

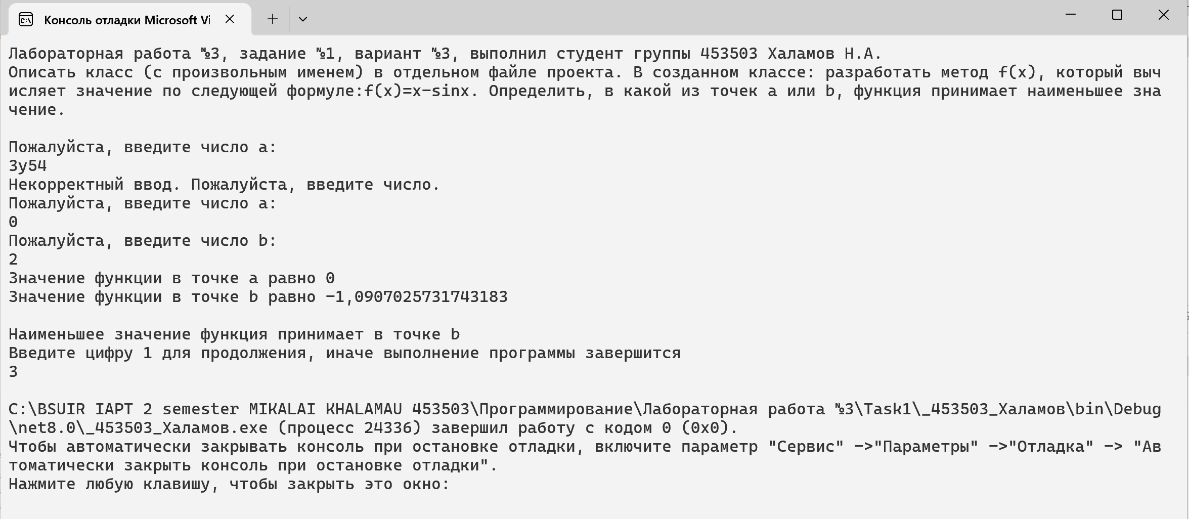


Рисунок 2 – Результат работы программы первого задания

Для второго задания были созданы 4 класса для вывода консольного меню с информацией о задании, для ввода чисел и проверки корректности ввода, для метода, вычисляющего выражение по условию, для всех функции в общем и возможности повторного ввода значений в консоль. Были использованы методы класса Math: Sin(для вычисления синуса), Log(для вычисления натурального логарифма), Exp(для возведения в степень числа е), Pow(для возведения числа в нужную степень). Были использованы операторы if else и switch.

using System;

namespace \_453503\_Халамов

{

internal class Method

{

public static void Function(double k, double z)

{

double x;

if (k < 1)

{

Console.WriteLine("Вычисление производилось по первой ветке");

x = k \* Math.Pow(z, 3);

}

else

{

Console.WriteLine("Вычисление производилось по второй ветке");

x = z \* (z + 1);

}

double a = Math.Log(1 + x \* x)+Math.Cos(1+x);

double b = Math.Exp(k \* x);

double y = Math.Pow(a, b);

Console.WriteLine($"Значение функции равно {y}");

}

}

}

Был продемонстрирован результат работы программы (см. рисунок 3).

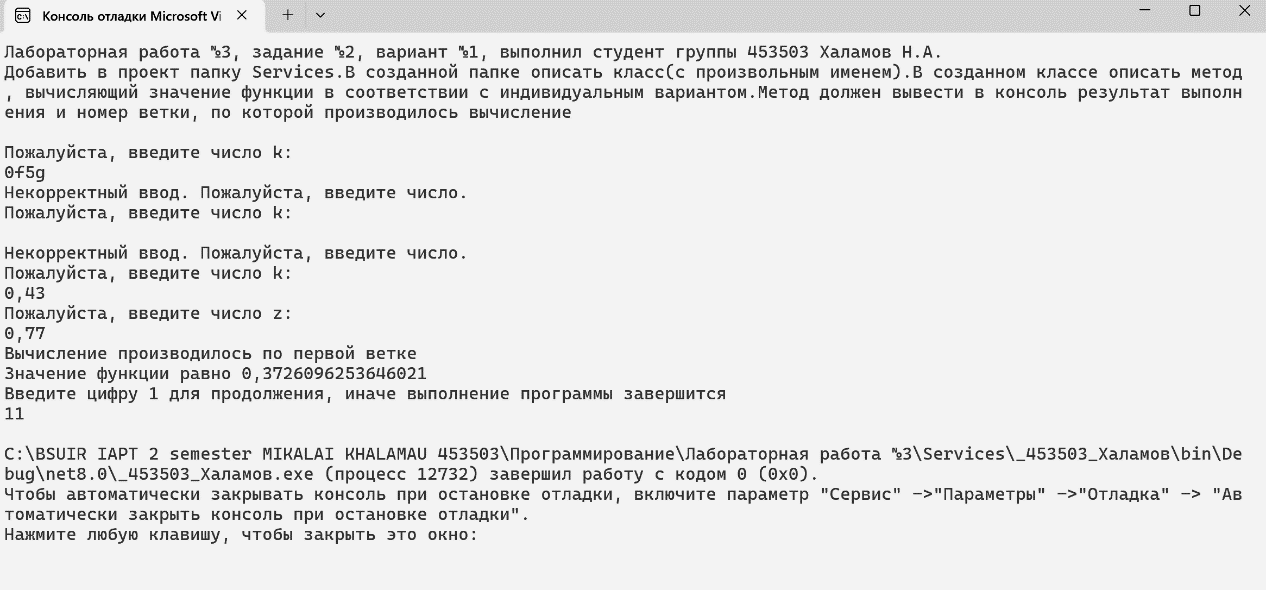


Рисунок 3 – Результат работы программы второго задания

В коде третьего задания используется пространство имен System.Globalization, которое предоставляет функциональность для глобализации и локализации приложения. Это позволяет нам учитывать особенности при работе с числами, строками и датами. При помощи класса DateTime были написаны методы для нахождения дня недели по введённой дате и разницы между двумя датами, выраженной в днях. Также реализована проверка на корректность ввода всех дат, учитывая високосность и количество дней в месяцах, а также максимальный возможный номер года.

using System;

using System.Globalization;

namespace \_453503\_Халамов

{

internal class DataService

{

public static void GetDaysSpan(int year, int month, int day)

{

DateTime date1 = new DateTime(year, month, day);

DateTime date2 = DateTime.Now;

Console.WriteLine($"Между датами пройдет {date1.Subtract(date2)}");

}

public static void GetDayOfWeek(string sdate)

{

DateTime dateTime = DateTime.Parse(sdate, CultureInfo.InvariantCulture);

CultureInfo culture = new CultureInfo("ru-RU");

Console.WriteLine($"День недели {sdate} : {dateTime.ToString("dddd", culture)}");

}

}

}

Был продемонстрирован результат работы программы (см. рисунок 4).

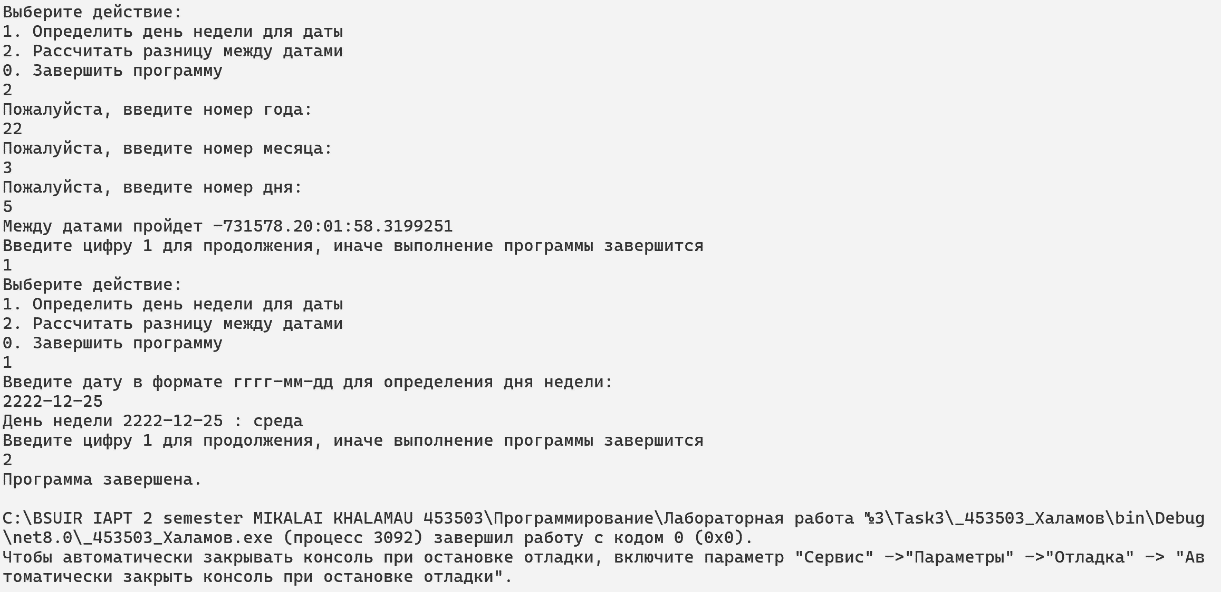


Рисунок 4 – Результат работы программы третьего задания

# Вывод

В ходе выполнения индивидуального задания были разработаны три программы, каждая из которых реализует определенную функциональность. В первой программе вычислялись значения функции f(x)=x−sin(x) в двух заданных точках, после чего определялось, в какой из них функция принимает наименьшее значение. Для этого были созданы отдельные классы для обработки ввода, вывода и вычислений, а также использован метод Math.Sin и конструкция if-else. Вторая программа, размещенная в папке Services, рассчитывала значение выражения по двум различным веткам в зависимости от входных параметров, используя Math.Sin, Math.Log, Math.Exp и Math.Pow, а также выводила в консоль, по какой ветке производилось вычисление. В третьей программе, работающей с датами, использовалось пространство имен System.Globalization и класс DateTime для определения дня недели по введенной дате и расчета разницы между заданной и текущей датой в днях. Все программы были структурированы в отдельных классах, что улучшило читаемость кода и обеспечило удобство его использования. Включение механизма повторного ввода позволило пользователю легко взаимодействовать с программой. Были разработаны проверки корректности ввода всех значений дат. Проведенные тестирования подтвердили корректность работы всех программ, а результаты выполнения были продемонстрированы графически.