Министерство образования Республики Беларусь

Учреждение образования «Белорусский государственный университет

информатики и радиоэлектроники»

Факультет компьютерного проектирования

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 34

|  |
| --- |
| Выполнил студент группы 353505  Хорошко Кирилл Николаевич |
|  |
| (дата, подпись студента) |
| Проверил ассистент каф. Информатики  РОМАНЮК Максим Валерьевич |
|  |
| (дата, подпись преподавателя) |

Минск 2024

# 1 Индивидуальное задание

**Задание 1. Вариант 6.** Разработать метод , который для заданного натурального числа и вещественного находит значение выражения . Вычислить с помощью данного метода значение выражения .

**Задание 1. Вариант 10.** Добавить в проект папку Services. В созданной папке описать класс (с произвольным именем). В созданном классе описать метод, вычисляющий значение функции на рисунке 1. Метод должен вывести в консоль результат выполнения и номер ветки, по которой производилось вычисление.

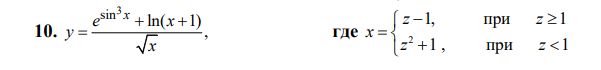


Рисунок 1 – Функция, значение которой необходимо вычислить

**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 выполнения данной лабораторной работы первым делом был создан проект, к которому позже добавлялись классы, реализующие отдельные задания. После реализации всех трех заданий в основном файле был написан код, запрашивающий номер выполняемого задания и в случае, если был введён номер существующего задания, входные данные (рисунок 2). Если было введен номер несуществующего задания, программы сообщала об этом и запрашивала данные повторно (рисунок 3). Это было реализовано с помощью конструкции switch/case. Так же в этом файле были реализованы бесконечный цикл для повторного ввода и проверка входных данных.

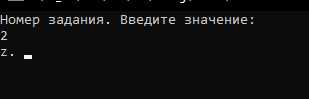


Рисунок 2 – результат обработки программой корректных данных, введённых после запроса программой номера задания

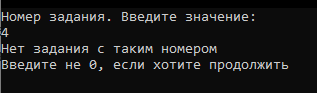


Рисунок 3 – результат обработки программой некорректных данных, введённых после запроса программой номера задания

Задание 1. Вариант 6.

В задании был создан класс, который содержал метод, в котором была определена функция заданная по условию задания.

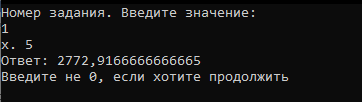


Рисунок 4 – результат работы программы при выборе задания 1 и вводе в качестве входных данных числа 5.

static public class Task1

{

public static double func(uint n, double x)

{

double ans = 1;

for (int i = 0; i < n; i++)

{

ans = ans \* x;

}

return ans /= n;

}

}

Задание 2. Вариант 4.

Для выполнения задания 2 была создана папка, в которой был создан класса, содержащий метод, который возвращает значение выражения из задания, а так же номер ветки, по которой проводились расчеты. Возврат нескольких значений реализовывался с помощью модификатора out.

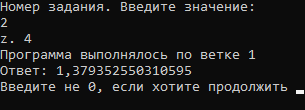


Рисунок 5 – результат работы программы при выборе задания 2 и вводе в качестве входных данных числа 4

static public class Task2

{

static public int func(double z, out double y)

{

double x;

int i;

if (z >= 1)

{

x = z - 1;

i = 1;

}

else

{

x = z \* z + 1;

i = 2;

}

y = (Math.Pow(Math.E, Math.Pow(Math.Sin(x), 3)) + Math.Log(x + 1)) / Math.Sqrt(x);

return i;

}

}

Задание 3.

В отдельном файле был создан класс, содержащий 2 метода: GetDay, который возвращает день недели в некоторую дату, и GetDaysSpan, который вычисляет сколько дней осталось или прошло от сегодняшнего дня до определенной даты.

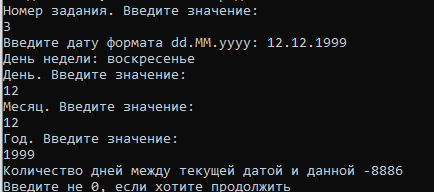


Рисунок 6 – результат работы программы при выборе задания 3 вводе в качестве входных параметров для обоих методов даты 12.12.1999

class Task3

{

static public class DataServis

{

static public string GetDay(string date)

{

string ans;

DateTime parse = DateTime.Parse(date);

ans = parse.ToString("dddd");

return ans;

}

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

{

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

return (usr.Date - DateTime.Today).Days;

}

}

}

**ВЫВОД**

В ходе работы было проведено ознакомление с методами классов Math , DateTime и TimeSpan языка C#. Также, были закреплены навыки управления потоком выполнения программы с использованием конструкций if else, switch case и тернарного оператора. Также, было осуществленно правильное разбиение программы на классы и методы, в соответствии с принципами ООП.