**1 Принципы объектно-ориентированного программирования**

Задание 1. Написать программу, которая вычисляет значение функции у:

Листинг программы:

using System;

using System.Linq;

class Program

{

static void Main()

{

double x, y;

for (x = 1; x <= 5; x += 0.1)

{

y = Math.Log(x) + Math.Pow(Math.Cos(x \* x), 2);

y = Math.Round(y, 1);

Console.WriteLine($"При x = {x}, y = {y}");

}

x = Math.PI;

y = Math.Pow(Math.Sin(x), 2);

Console.WriteLine($"При x = {x}, y = {y}");

}

}

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 1, 5, | 0.2…2.5 |
| 2, | 1,4997597826618576E-32 |

Анализ результатов:

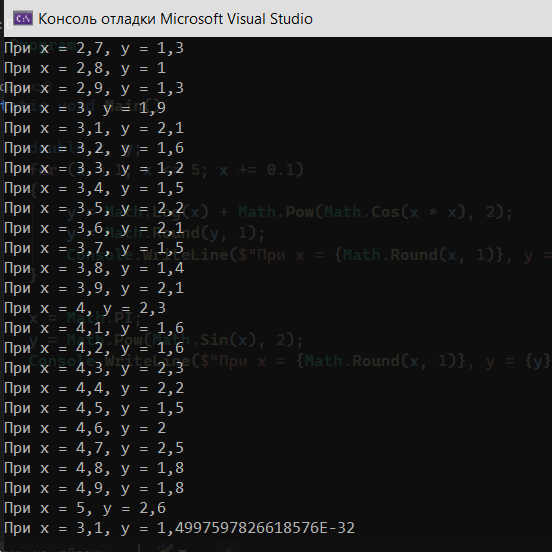


Рисунок 1.1 – Результат работы программы

Задание 2. Написать программу, которая определит какая из цифр трехзначного числа больше: первая или последняя.

Листинг программы:

using System;

class Program

{

static void Main()

{

int number = 123;

string numberStr = number.ToString();

int firstDigit = int.Parse(numberStr[0].ToString());

int lastDigit = int.Parse(numberStr[numberStr.Length - 1].ToString());

if (firstDigit > lastDigit)

{

Console.WriteLine("Первая цифра больше последней.");

}

else if (firstDigit < lastDigit)

{

Console.WriteLine("Последняя цифра больше первой.");

}

else

{

Console.WriteLine("Первая и последняя цифры равны.");

}

}

}

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 123 | Последняя цифра больше первой. |

Анализ результатов:

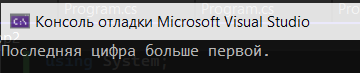


Рисунок 1.2 – Результат работы программы

Задание 3. Составить программу, которая решит задачу.

Во время сильного дождя на остановке стояло 12 человек. Подкативший автобус забрызгал грязью 4-го, 5-го, 6, 7 и 8-го человека. Остальные попрыгали в колючие кусты, причем 3-ий, 9-ый и 12-ый так и не смогли выбраться из них. Задать номер пассажира и определить грязный он или исцарапанный.

Листинг программы:

using System;

using System.Collections.Generic;

class Program

{

static void Main()

{

List<int> dirtyPeople = new List<int> { 4, 5, 6, 7, 8 };

List<int> scratchedPeople = new List<int> { 3, 9, 12 };

Console.Write("Введите номер пассажира: ");

int passengerNumber = Convert.ToInt32(Console.ReadLine());

if (dirtyPeople.Contains(passengerNumber))

{

Console.WriteLine("Пассажир номер " + passengerNumber + " грязный.");

}

else if (scratchedPeople.Contains(passengerNumber))

{

Console.WriteLine("Пассажир номер " + passengerNumber + " исцарапан.");

}

else

{

Console.WriteLine("Пассажир номер " + passengerNumber + " в безопасности.");

}

}

}

Таблица 1.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 4 | Пассажир номер 4 грязный. |

Анализ результатов:

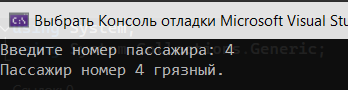


Рисунок 1.3 – Результат работы программы

Задание 4. Вывести все целые числа из диапазона от А до В, кратные трем ( AB)

Листинг программы:

using System;

class Program

{

static void Main()

{

int A = 3;

int B = 30;

int i;

Console.WriteLine("Числа, кратные трем, используя цикл for:");

for (i = A; i <= B; i++)

{

if (i % 3 == 0)

{

Console.WriteLine(i);

}

}

i = A;

Console.WriteLine("Числа, кратные трем, используя цикл while:");

while (i <= B)

{

if (i % 3 == 0)

{

Console.WriteLine(i);

}

i++;

}

i = A;

Console.WriteLine("Числа, кратные трем, используя цикл do-while:");

do

{

if (i % 3 == 0)

{

Console.WriteLine(i);

}

i++;

} while (i <= B);

}

}

Таблица 1.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
|  | 3,6,9,12,15,18,21,24,27,30 |

Анализ результатов:

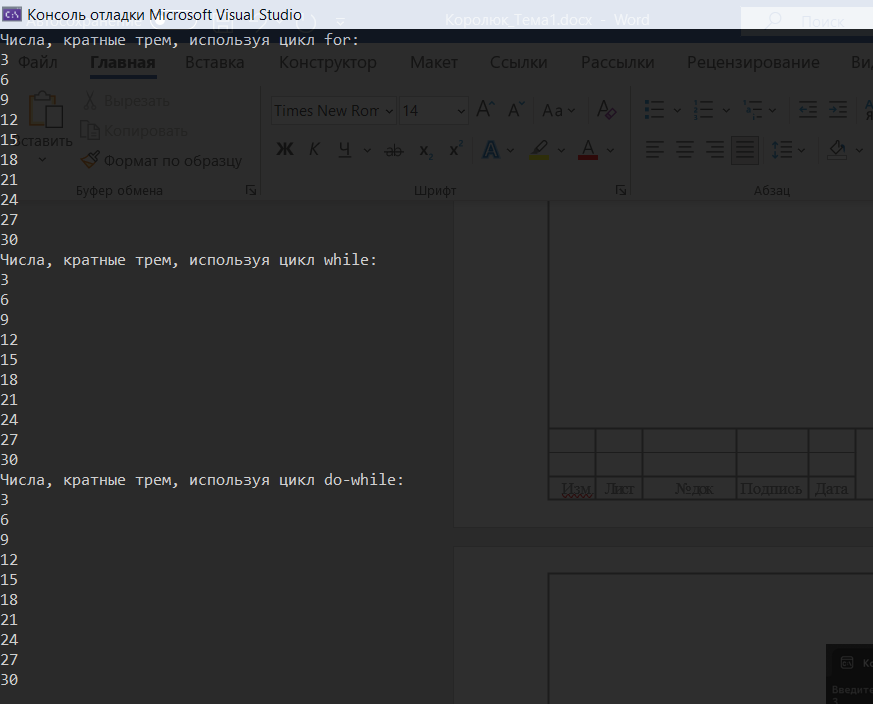


Рисунок 1.4 – Результат работы программы

Задание 5. Даны два целых числа A и B (A < B). Найти сумму квадратов всех

целых чисел от A до B включительно.

Входные данные: ввести два целых числа A, B (1<=A,B<=10).

Выходные данные: вывести сумму квадратов всех целых чисел от A до B включительно.

Листинг программы:

using System;

class Program

{

static void Main()

{

Console.Write("Введите число A: ");

int A = Convert.ToInt32(Console.ReadLine());

Console.Write("Введите число B: ");

int B = Convert.ToInt32(Console.ReadLine());

int sum = 0;

for (int i = A; i <= B; i++)

{

sum += i \* i;

}

Console.WriteLine("Сумма квадратов всех целых чисел от A до B включительно: " + sum);

}

}

Таблица 1.5 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 5,10 | Сумма квадратов всех целых чисел от A до B включительно: 355 |

Анализ результатов:

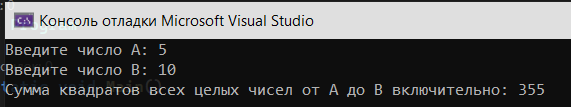


Рисунок 1.5 – Результат работы программы

Задание 6. Составить программу вычисления значений функции F(x) = sin() на отрезке [A, B], A = π /6, B = 2π / 3 в точках xi = x + H, где H=(B-A)/M, M = 10

Листинг программы:

using System;

class Program

{

static void Main()

{

double A = Math.PI / 6;

double B = 2 \* Math.PI / 3;

int M = 10;

double H = (B - A) / M;

Console.WriteLine("x\t\tsin(x^2)");

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

{

double x = A + H \* i;

double F = Math.Sin(Math.Pow(x, 2));

Console.WriteLine($"{x}\t\t{F}");

}

}

}

Таблица 1.6 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| π /6, 2π / 3, 10 | 0,2707342433100194, 0,4469233182543233,  0,6456227869247629,  0,8358624301944239,  0,9704076922538113,  0,9898389400522631,  0,8372377105940066,  0,4855188706628955,  -0,02764345998479067,  -0,5741769969009938,  -0,9473635617014299 |

Анализ результатов:

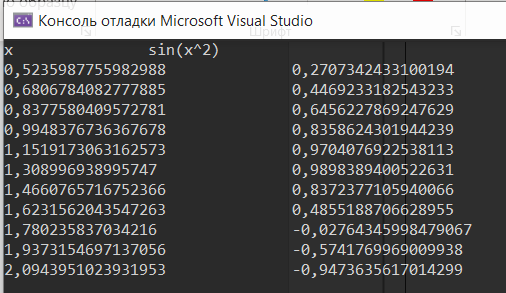


Рисунок 1.6 – Результат работы программы

Задание 7. Написать программу, определяющая сколько раз цифра k входит в десятичную запись заданного числа n.

Листинг программы:

using System;

class Program

{

static void Main()

{

Console.Write("Введите число n: ");

string n = Console.ReadLine();

Console.Write("Введите цифру k: ");

char k = Console.ReadKey().KeyChar;

int count = 0;

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

{

if (n[i] == k)

{

count++;

}

}

Console.WriteLine("\nЦифра " + k + " встречается в числе " + n + " " + count + " раз.");

}

}

Таблица 1.7 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| 356783, 3 | 3 |

Анализ результатов:

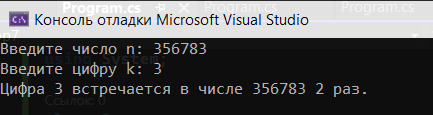


Рисунок 1.7 – Результат работы программы