**Принципы объектно-ориентированного программирования**

**Практическая часть**

**Задание №1.** Написать программу, которая вычисляет значение функции *у*:



Листинг программы:

static void Main(string[] args)

{

try

{

WriteLine("Введите число x: ");

double x = ToDouble(ReadLine());

double y;

if (x <= Math.PI)

{

y = x + 2 \* x \* Math.Sin(3 \* x);

}

else

{

y = Math.Cos(x) + 2;

}

WriteLine($"Y = {y}");

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadLine();

}

} }

Таблица 1.1 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| X = 5 | Y = 2.28 |

Анализ результатов:

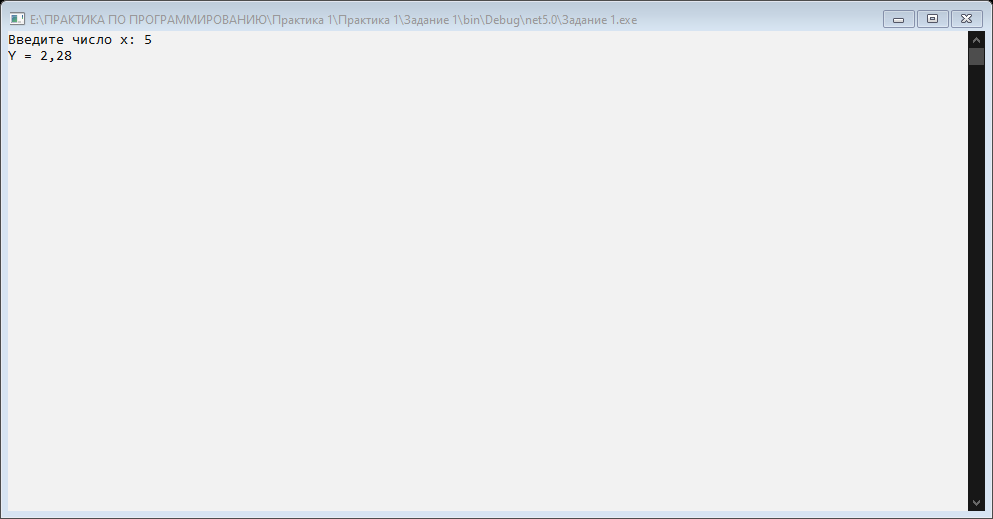


Рисунок 1.1 – Результат работы программы

**Задание №2.** Написать программу, которая определяет: существует ли треугольник с длинами сторон *a*, *b*, *c*.

Листинг программы:

static void Main(string[] args)

{

try

{

WriteLine("Введите стороны треугольника...");

WriteLine("Сторона А: ");

double a = ToDouble(ReadLine());

WriteLine("Сторона В: ");

double b = ToDouble(ReadLine());

WriteLine("Сторона С: ");

double c = ToDouble(ReadLine());

if (!(a + b > c) || !(b + c > a) || !(a + c > b))

{

WriteLine("Треугольник не существует...");

}

else

{

WriteLine("Треугольнеик существует...");

}

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadKey();

Таблица 1.2 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| A = 4  B = 6  C = 10 | Треугольник не существует. |

Анализ результатов:

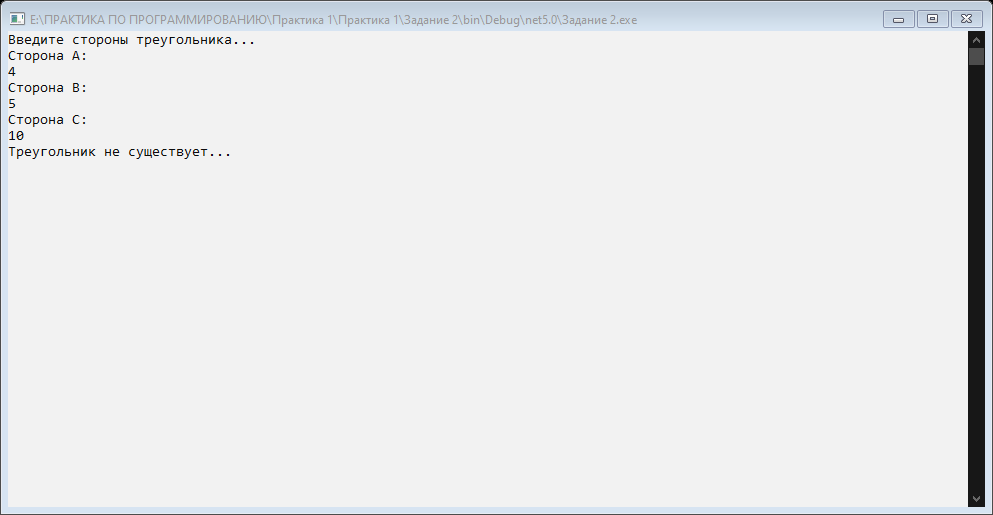


Рисунок 1.2 – Результат работы программы

**Задание №3**.Составить программу: Дан порядковый номер дня месяца, вывести на экран количество дней оставшихся до конца месяца.

Листинг программы:

public static void ReturnDays(int your\_day, int day\_count)

{

var answer = day\_count - your\_day;

WriteLine($"Дней до конца месяца: {answer}.");

}

static void Main(string[] args)

{

var Monthes = new Dictionary<string, int>()

{

["Январь"] = 31,

["Февраль"] = 28,

["Март"] = 31,

["Апрель"] = 30,

["Май"] = 31,

["Июнь"] = 30,

["Август"] = 31,

["Сентябрь"] = 30,

["Октрябрь"] = 31,

["Ноябрь"] = 30,

["Декабрь"] = 31

};

try

{

WriteLine("Выберите месяц: ");

string month = ReadLine();

WriteLine("Выберите день: ");

int day = ToInt32(ReadLine());

if (Monthes.ContainsKey(month))

{

ReturnDays(day, Monthes[month]);

}

else

{

WriteLine("Такого месяца не существует...");

}

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadKey();

}

Таблица 1.3 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Месяц: Март  День: 28 | Дней до конца месяца: 3 |

Анализ результатов:

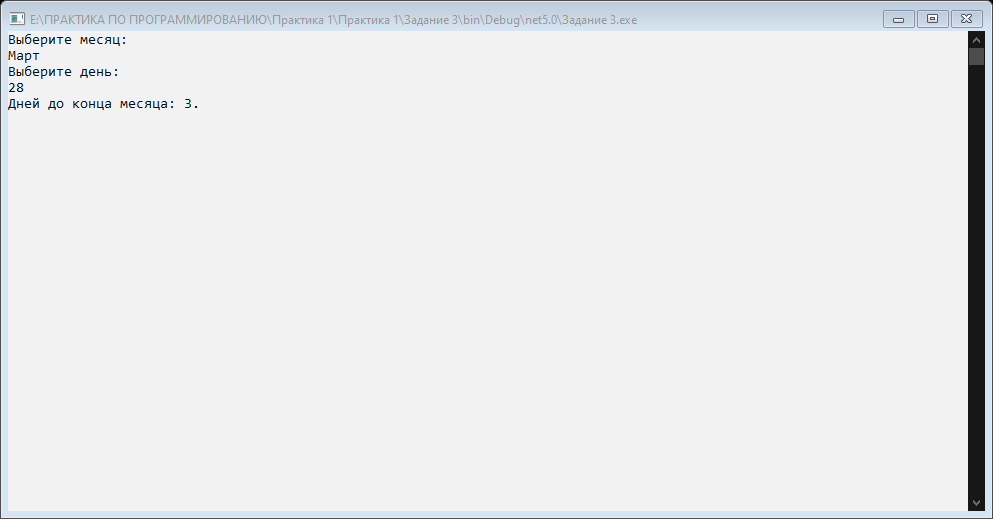


Рисунок 1.3 – Результат работы программы

**Задание №4.** Вывести на экран (задачу решите тремя способами - используя операторы цикла while, do while и for): аблицу перевода 5, 10, 15, …, 500 долларов США в рубли по текущему курсу (значение курса вводится с клавиатуры).

Листинг программы:

static void Main(string[] args)

{

try

{

WriteLine("Введите курс доллара: ");

double dollar\_rate = ToDouble(ReadLine());

double rubles;

WriteLine("----------------\n");

for (int i = 5; i <= 500; i += 5)

{

rubles = i \* dollar\_rate;

WriteLine($"{i}$ = {rubles} р. |");

}

WriteLine("----------------\n" +

"Цикл FOR\n");

int dollar = 5;

while (dollar <= 500)

{

rubles = dollar \* dollar\_rate;

WriteLine($"{dollar}$ = {rubles} р. |");

dollar += 5;

}

WriteLine("----------------\n" +

"Цикл WHILE \n");

int temp\_dollar = 5;

do

{

rubles = temp\_dollar \* dollar\_rate;

WriteLine($"{temp\_dollar}$ = {rubles} р. |");

temp\_dollar += 5;

} while (temp\_dollar <= 500);

WriteLine("----------------\n" +

"Цикл DO WHILE \n");

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadKey();

}

Таблица 1.4 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Курс доллара: 3,20 | 5$ = 16 р.  10$ = 32 р.  495$ = 1584 р. |  500$ = 1600 р. |

Анализ результатов:

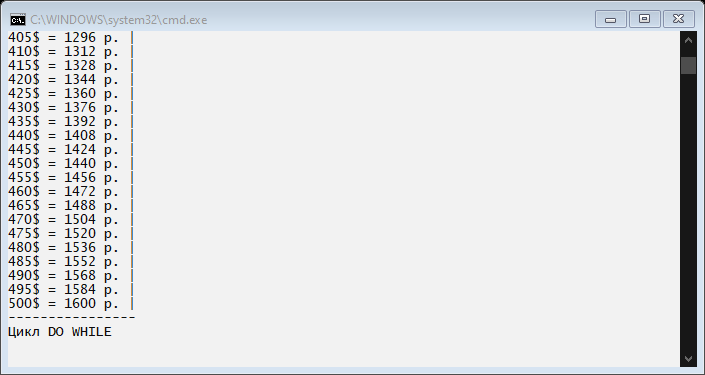


Рисунок 1.4 – Результат работы программы

**Задание № 5.** Дано целое число — цена 1 кг конфет. Вывести стоимость 1, 2…10 кг конфет.

Входные данные: ввести одно целое число A (1<=A<=100).

Выходные данные: вывести стоимость 1, 2, … , 10 кг конфет.

Листинг программы:

static void Main(string[] args)

{

try

{

WriteLine("Введите стоимость 1кг конфет: ");

double cost\_1kg = ToDouble(ReadLine());

if (cost\_1kg >= 1 && cost\_1kg <= 100)

{

for (int i = 1; i <= 10; i++)

{

WriteLine($"{i}кг. стоит {Math.Round((cost\_1kg \* i), 2)}р.");

}

}

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadKey();

}

Таблица 1.5 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| Стоимость 1кг: 3,15 | 1кг. стоит 3,15р.  2кг. стоит 6,3р.  9кг. стоит 28,35р.  10кг. стоит 31,5р. |

Анализ результатов:

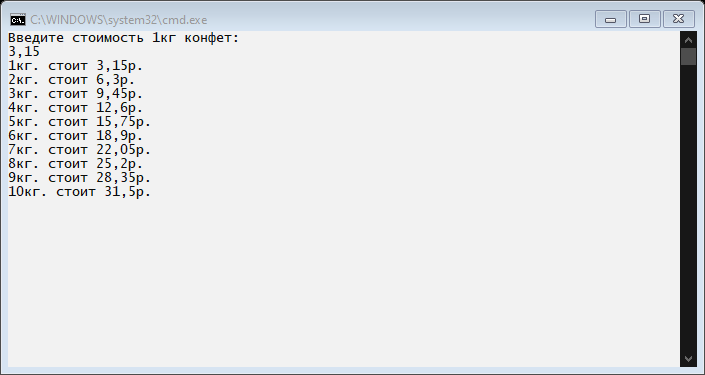


Рисунок 1.5 – Результат работы программы

**Задание № 6.** Составить программу вычисления значений функции F(x) на отрезке [A, B] в точках xi=x+H, где H=(B-A)/M, M – заданное целое число.

Листинг программы:

static void Main(string[] args)

{

const int M = 20;

const int A = 0;

const double B = Math.PI / 2;

const double n = 2.1;

const double e = 2.71;

try

{

Write("Введите значение x: ");

double x = ToDouble(ReadLine());

x = Math.Sin(x) - Math.Cos(x);

double H = (B - A) / M;

for (double i = 0.1; i <= n; i += 0.1)

{

double y = Math.Pow(x, 2) - Math.Pow(e, x);

x = x + H;

WriteLine($"Y = {Math.Round(y,2)}");

}

}

catch (Exception ex)

{

WriteLine(ex.Message);

}

finally

{

ReadKey();

}

Таблица 1.6 – Входные и выходные данные

|  |  |
| --- | --- |
| Входные данные | Выходные данные |
| М = 20  А = 0  В = p/2  N = 2,1  E = 2,71  X = 5 | Y = 1,25  Y = 1,04  Y = -1,16  Y = -1,22 |

Анализ результатов:

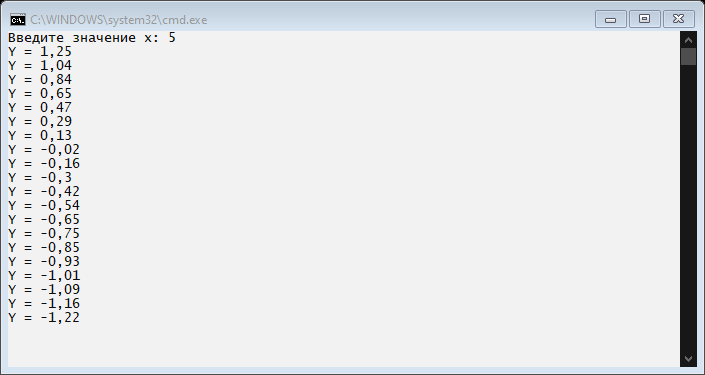


Рисунок 1.6 – Результат работы программы