**Лабораторная работа №1**

**Создание, компилирование, отладка и выполнение проектов в ИСР**

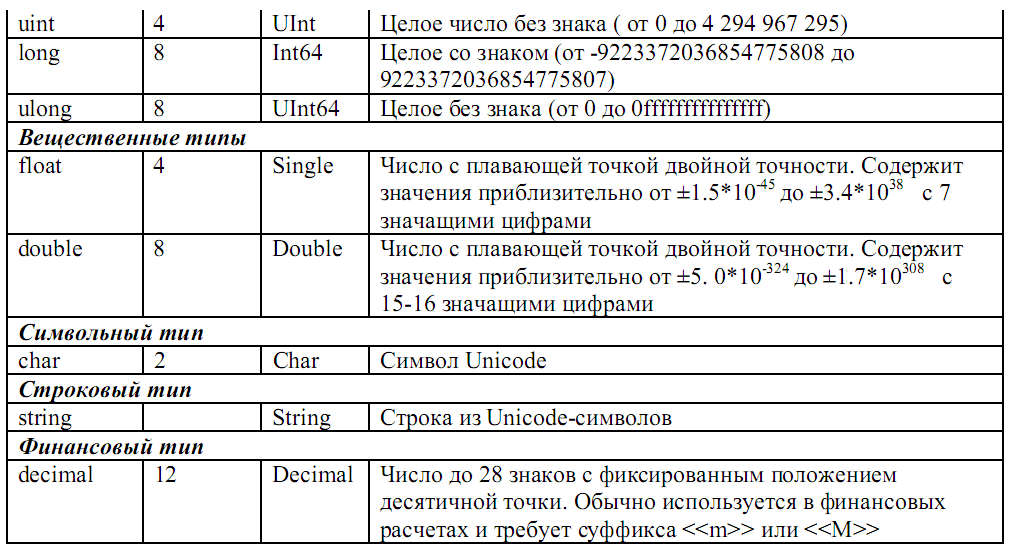
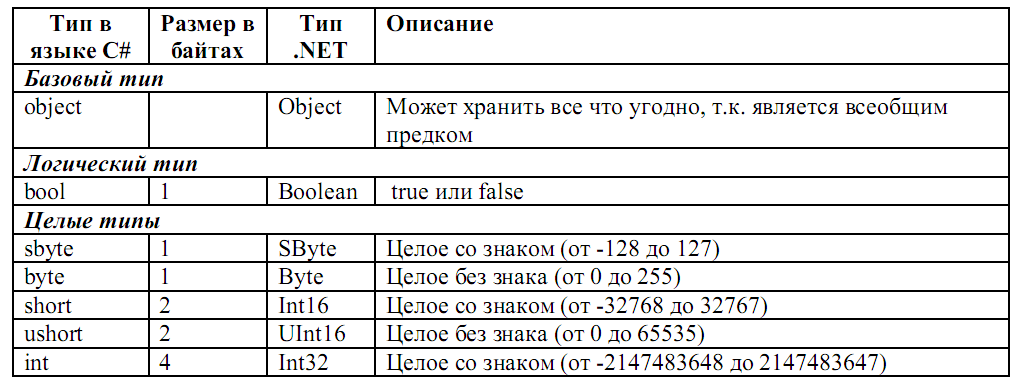
**Microsoft Visual Studio .NET. Разработка линейных программ.**

**Цель работы:**

Познакомиться с вводом-выводом данных, научиться использовать их в программах с использованием среды **Visual Studio.Net.**

**1. Теоретические сведения**

Язык С# предлагает обычный набор базовых типов, каждому из них соответствует тип, поддерживаемый общеязыковой спецификацией .NET (CLS).



Программа при вводе данных и выводе результатов взаимодействует с внешними устройствами. Совокупность стандартных устройств ввода (клавиатура) и вывода (экран) называется консолью. В языке С# нет операторов ввода и вывода.

Вместо них для обмена данными с внешними устройствами используются специальные классы. В частности, для работы с консолью используется стандартный класс Console, определенный в пространстве имен System.

**Вывод данных**

Метод WriteLine, реализованный в классе Console, позволяет организовывать вывод данных на экран. Существует несколько способов применения данного метода:

1. Console.WriteLine(x); //на экран выводится значение идентификатора х

2. Console.WriteLine("x=" + x +"y=" + y); /\* на экран выводится строка, образованная последовательным слиянием строки "x=", значения x, строки "у=" и значения у \*/

3. Console.WriteLine("x={0} y={1}", x, y); /\* на экран выводится строка, формат которой задан первым аргументом метода, при этом вместо параметра {0} выводится значение x, а вместо {1} – значение y\*/

Данный вариант использования метода WriteLine является наиболее универсальным, потому что он позволяет не только выводить данные на экран, но и управлять форматом их вывода.

*Управление размещением вещественных данных:*

Первым аргументом WriteLine указывается строка вида {n: ##.###} – где n определяет номер идентификатора из списка аргументов метода WriteLine, а ##.### определяет формат вывода вещественного числа. В данном случае, под целую часть числа отводится две позиции, под дробную – три. Если выделенных позиций для размещения целой части значения идентификатора окажется недостаточно, то автоматически добавится необходимое количество позиций.

Пример:

static void Main()

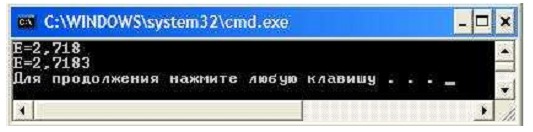
{

double x= Math.E;

Console.WriteLine("E={0:##.###}", x);

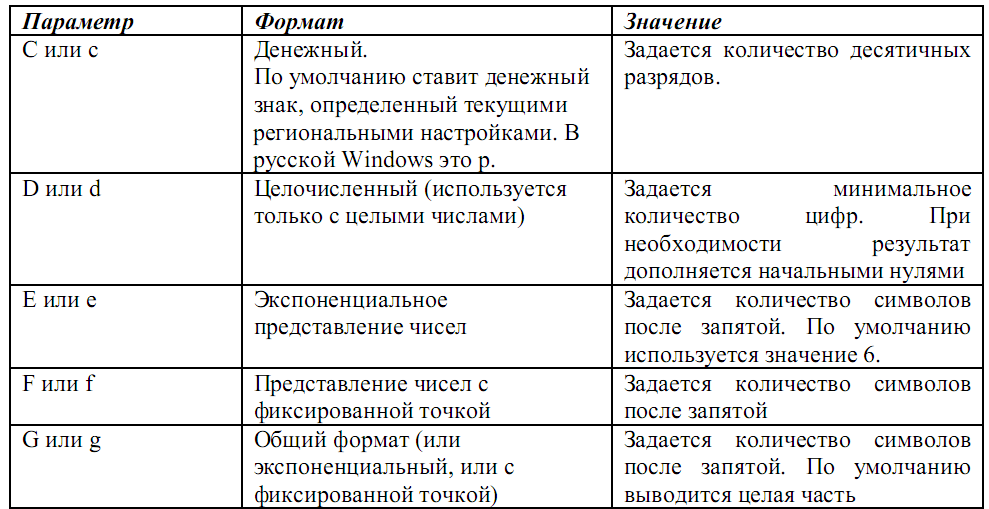
Console.WriteLine("E={0:.####}", x);

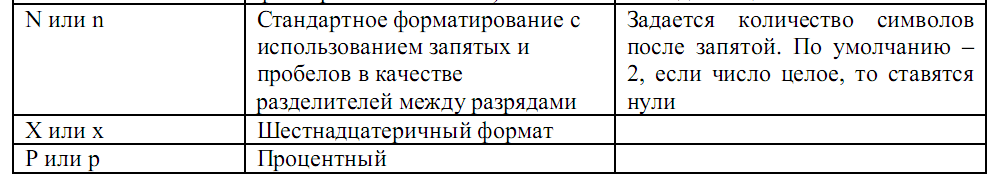
}

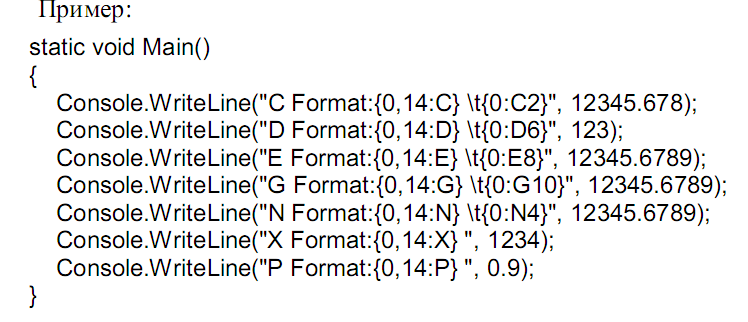


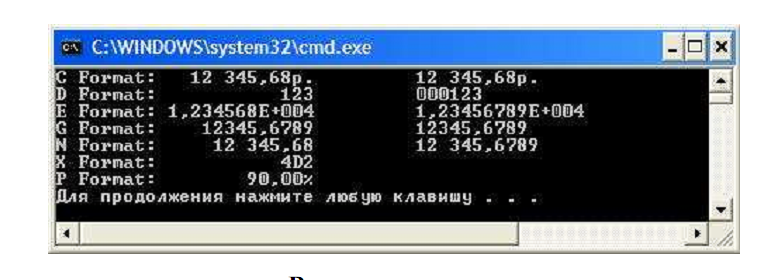
*Управление форматом числовых данных:*

Первым аргументом WriteLine указывается строка вида {n:<спецификатор>m} – где n определяет номер идентификатора из списка аргументов метода WriteLine, <спецификатор> - определяет формат данных, а m – количество позиций для дробной части значения идентификатора. В качестве спецификаторов могут использоваться следующие значения:









**Ввод данных**

Для ввода данных обычно используется метод ReadLine, реализованный в классе Console. Данный метод в качестве результата возвращает строку, тип которой string.

Пример:

static void Main()

{

string s = Console.ReadLine();

Console.WriteLine(s);

}

Для того чтобы получить числовое значение, необходимо воспользоваться преобразованием данных. Пример:

static void Main()

{

string s = Console.ReadLine();

int x = int.Parse(s); //преобразование строки в число

Console.WriteLine(x);

}

Или сокращенный вариант:

static void Main()

{ //преобразование введенной строки в число

int x = int.Parse(Console.ReadLine());

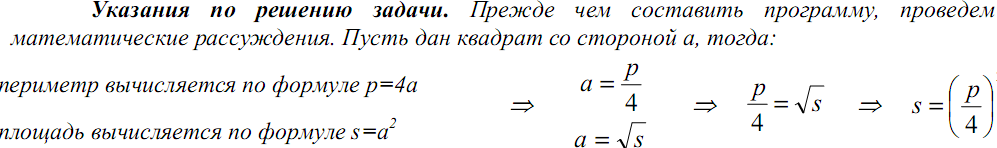
Console.WriteLine(x);

}

Для преобразования строкового представления целого числа в тип int мы используем метод Parse(), который реализован для всех числовых типов данных. Таким образом, если нам потребуется преобразовать строковое представление в вещественное, мы можем воспользоваться методом float.Parse() или double.Parse(). В случае, если соответствующее преобразование выполнить невозможно, то выполнение программы прерывается и генерируется исключение. Например, если входная строка имела неверный формат, то будет сгенерировано исключение System.FormatExeption.

**Примеры решения практических задач**

1. Написать программу, подсчитывающую площадь квадрата, периметр которого равен p.



using System;

namespace Example

{

class Program

{

static void Main()

{

Console.Write("p= ");

double p = double.Parse(Console.ReadLine());

double s = Math.Pow(p/4, 2);

Console.WriteLine("s{0},=" s);

}

}

}

Результат работы программы:

р=20

s=25

**Задание**. Изменить программу так, чтобы она подсчитывала периметр квадрата, площадь которого равна s.

2. Определить, является ли сумма цифр натурального двухзначного числа четной.

***Указание по решению задачи.*** *Напомним, что число является четным, если остаток от деления данного числа на 2 равен нулю. А для того, чтобы разложить двухзначное число на цифры, нужно разделить его нацело на 10 – так мы найдем старшую цифру числа, а затем взять остаток от деления на 10 исходного числа – так мы найдем его младшую цифру.*

using System;

namespace Example

{

class Program

{

static void Main()

{

Console.Write("a= ");

byte a = byte.Parse(Console.ReadLine());

string result=((a/10+a%10)%2==0)? "четное": "нечетное";

Console.WriteLine(result);

}

}

}

Результат работы программы:

х Сообщение на экране

45 нечетное

88 четное

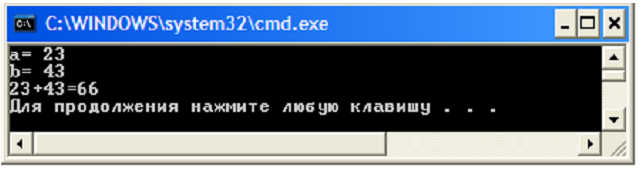
55 четное

**Задание**. Изменить программу так, чтобы она определяла, является ли сумма цифр трехзначного числа нечетной.

**Задания для самостоятельной работы**

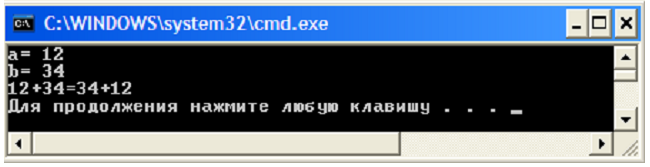
I. Написать программу, которая, реализует диалог с пользователем:

1) запрашивает с клавиатуры два целых числа, и выводит на экран сумму данных чисел:

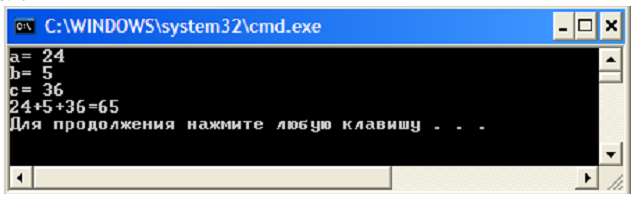


2) запрашивает с клавиатуры два целых числа, и выводит на экран сумму данных чисел

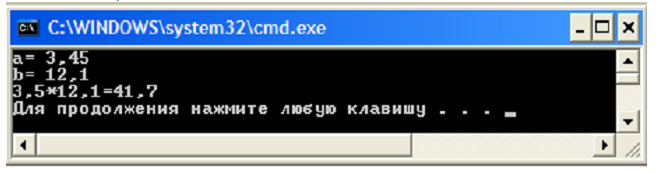
в прямом и обратном порядке:



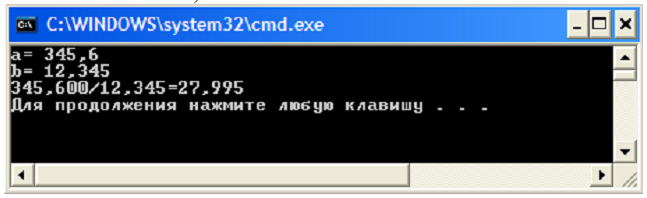
3) запрашивает с клавиатуры три целых числа, и выводит на экран сумму данных чисел:



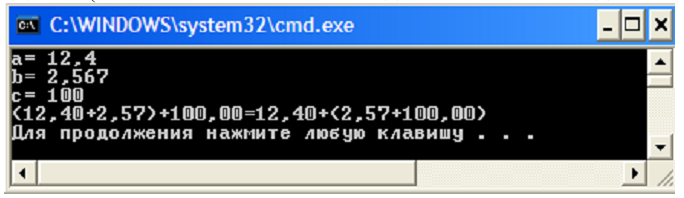
4) запрашивает с клавиатуры два вещественных числа, и выводит на экран произведение данных чисел (вещественные числа выводятся с точностью до 1 знака после запятой):



5) запрашивает с клавиатуры два вещественных числа, и выводит на экран результат деления первого числа на второе (вещественные числа выводятся с точностью до 3 знаков после запятой):



6) запрашивает с клавиатуры три вещественных числа, и выводит на экран следующее сообщение (вещественные числа выводятся с точностью до 2 знаков после запятой):



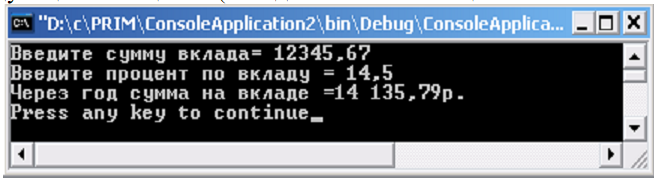
7) запрашивает с клавиатуры номинал купюры и количество купюр, и выводит экран следующее сообщение:



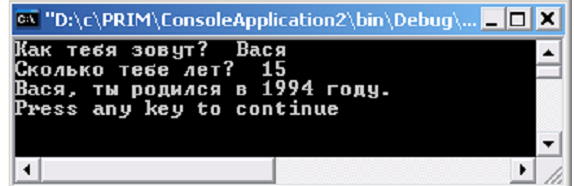
8) запрашивает с клавиатуры сумму вклада и процент по вкладу, и выводит на экран следующее сообщение (вклад без капитализации – все начисления в конце года):



9) запрашивает с клавиатуры сумму вклада и процент по вкладу, и выводит на экран следующее сообщение (вклад без капитализации – все начисления в конце года):



10) запрашивает с клавиатуры имя человека и его возраст, и выводит на экран следующее сообщение (текущий год – 2020)



II. Написать программу, которая подсчитывает:

1) площадь равностороннего треугольника, периметр которого равен p;

2) расстояние между точками с координатами x1, y1 и x2,y2;

3) гипотенузу прямоугольного треугольника по двум данным катетам a, b.

4) площадь прямоугольного треугольника по двум катетам a, b.

5) периметр прямоугольного треугольника по двум катетам a, b.

6) ребро куба, площадь полной поверхности которого равна s;

7) ребро куба, объем которого равен v;

8) периметр треугольника, заданного координатами вершин x1, y1, x2, y2, x3, y3;

9) площадь треугольника, заданного координатами вершин x1, y1, x2, y2, x3, y3;

10) сумму членов арифметической прогрессии, если известен ее первый член,

разность и число членов прогрессии;

11) площадь квадрата, заданного координатами противоположных вершин x1, y1, x2, y2 (стороны квадрата не параллельны осям координат);

12) площадь квадрата, заданного координатами противоположных вершин x1, y1, x2, y2 (стороны квадрата параллельны осям координат);

13) площадь квадрата, заданного координатами соседних вершин x1, y1, x2, y2 (стороны квадрата не параллельны осям координат).

III. Написать программу, которая определяет:

1) наибольшую цифру в натуральном двухзначном числе;

2) наименьшую цифру в натуральном двухзначном числе;

3) одинаковы ли цифры данного двухзначного числа;

4) заканчивается ли сумма цифр двухзначного числа на 0;

5) кратна ли трем сумма цифр двухзначного числа;

6) кратна ли числу А сумма цифр двухзначного числа;

7) какая из цифр трехзначного числа больше: первая или последняя;

8) какая из цифр трехзначного числа больше: первая или вторая;

9) какая из цифр трехзначного числа больше: вторая или последняя;

10) все ли цифры трехзначного числа одинаковые;

11) заканчивается ли сумма цифр трехзначного числа на 2;

12) наименьшую цифру в натуральном трехзначном числе;

13) наибольшую цифру в натуральном трехзначном числе.