



Программирование на C#

Семинар №8

Модуль №1

Тема:

Передача параметров в методы.
Решение задач с out и ref.



Задания преподавателя к семинару

- Не забываем создавать методы в отдельном статическом классе. Класс размещать в отдельном файле проекта.
- Для всех решений предлагать повторный ввод данных, если они некорректно указаны пользователем
- Для всех задач использовать цикл повторения решения.
- Выполнить задания категорий ToDo и Self.



Полезные материалы к семинару

1. out (Справочник по C#) <https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/out?redirectedfrom=MSDN>
2. Ссылка (справочник по C#) <https://learn.microsoft.com/ru-ru/dotnet/csharp/language-reference/keywords/ref>
3. Статья из справочника по языку C# (<https://msdn.microsoft.com/ru-ru/library/t3c3bfhx.aspx>)
4. Статья «Модификатор параметров out» (<https://msdn.microsoft.com/ru-ru/library/ee332485.aspx>)

Demo 01.



```
int s1 = 0;
Console.WriteLine("  Main s1 {0}", s1);
int a = 2, b = 3;
p_ref(a, b, ref s1);
Console.WriteLine(" \n Опять Main  a {0} b {1} s1 {2} ", a, b, s1);

static void p_ref(int a1, int b1, ref int s)
{
    s = a1 + b1;
    Console.WriteLine("\n p_ref ");
    Console.WriteLine(" a1 {0} b1 {1}", a1, b1);
    Console.WriteLine(" Сумма {0} ", s);
}
```

Если s1 не присвоить значение до вызова метода с ref, будет ошибка!

ToDo 01. Определите вывод значения arg



```
static void Main(string[] args)
{
    int arg;
    arg = 4;
    squareVal(arg);
    Console.WriteLine(arg);
    arg = 4;
    squareRef(ref arg);
    Console.WriteLine(arg);
}
```

ссылка: 1

```
static void squareVal(int valParameter)
{
    valParameter *= valParameter;
}
```

ссылка: 1

```
static void squareRef(ref int refParameter)
{
    refParameter *= refParameter;
}
```





Модификатор out

1. Параметры с модификатором **out** также называют «выходными» [out parameters].
2. Статья из справочника по языку C# (<https://msdn.microsoft.com/ru-ru/library/t3c3bfhx.aspx>)
3. Статья «Модификатор параметров out» (<https://msdn.microsoft.com/ru-ru/library/ee332485.aspx>)

Формулы вам пригодятся для выполнения задания **Self01**

Формула приближённого вычисления π с использованием кратных рядов

Формула приближённого вычисления π
Бэйли-Боруэйна-Плаффа

$$\pi = \sum_{i=0}^{\infty} 16^{-i} \left(\frac{4}{8i+1} - \frac{2}{8i+4} - \frac{1}{8i+5} - \frac{1}{8i+6} \right)$$

$$\pi = \sqrt[4]{360 \sum_{k=1}^{\infty} \sum_{m=1}^k \frac{1}{m(k+1)^3}}$$

Self01.Площадь и Периметр Треугольника



1. В классе Program, размещённом в файле Program.cs описать статические методы, вычисляющие приближённые (с машинной точностью) значения числа π по формуле: Бэйли-Боруэйна-Плаффа и с кратными рядами

Оба метода возвращают вычисленное значение числа π и количество итераций, за которые достигнута требуемая точность.

2. В том же классе разместить код метода Main(), который:
 - Предлагает пользователю текстовое меню для выбора одного из двух методов приближённого вычисления числа π .
 - На основе выбранного пункта запускает один из методов и формировать вывод, содержащий приближённое значение числа π и количество итераций, затраченных на получения значения с машинной точностью
 - Организует цикл повторения решения, позволяющий выполнить шаги 1-5. Выполнение повторений решения программы заканчивать после нажатия клавиши **Escape**.

Ограничения и требования:

1. Придерживайтесь рекомендаций по комментированию кода программы.
2. Постарайтесь в вычислениях минимизировать количество циклов.

Self02. Площадь и Периметр Треугольника



Напишите метод, вычисляющий площадь (**a**) и периметр (**p**) треугольника по заданным длинам его сторон. Метод должен возвращать значение **false**, если параметры заданы неверно и треугольник с такими длинами построить нельзя или **true** в случае успеха.

В основной программе вводятся значения трёх вещественных чисел – сторон треугольника.

Заголовок метода:

```
public static bool Triangle(double x, double y, double z,  
    out double p, out double s)
```




Self 03. Циклический Сдвиг Символа

Напишите метод, изменяющий значение символьного параметра русского или английского алфавита путём его циклического сдвига на `shiftCount` позиций вперёд или назад (если значение `shiftCount` отрицательное).

Метод должен возвращать значение `false`, если аргумент не является символом русского или английского алфавита, и `true` в случае успеха.

В основной программе вводится символ и количество позиций, на которое выполняется его циклический сдвиг с помощью описанного метода. Если метод вернул `false`, символ должен остаться неизменным.

Заголовок метода:

```
public static bool Shift(int shiftCount, ref char ch)
```

Примеры сдвига: (b, 4) → f; (Г, 10) → М; (V, -3) → S; (d, 26) → d;
(Z, 53) → A



Self 04. Много маленьких задач

1. Напишите метод **SummIt()**, возвращающий количество натуральных чисел из диапазона **[A, B]**, в записи которых содержится хотя бы одна единица и сумму этих чисел, **A, B** — вещественные параметры метода. Опишите метод **TotalSumm()**, возвращающий сумму всех натуральных чисел из диапазона **[A, B]**, **A, B** — вещественные параметры метода. В основной программе получать от пользователя два вещественных числа — границы диапазона и выводить модуль разности сумм всех чисел и чисел, в записи которых содержится хотя бы одна единица.
2. Написать метод **InvertDigits(K)**, возвращающий **void** и меняющий порядок следования цифр целого положительного числа **K** на обратный (**K** — параметр целого типа изменяется в теле метода). С помощью этой метода поменять порядок следования цифр на обратный введенного пользователем с клавиатуры в вызывающей программе целого числа.
3. Написать метод **Minmax(X, Y)**, возвращающий значение типа **void** и записывающий в параметр **X** минимальное из значений **X** и **Y**, а в параметр **Y** — максимальное из этих значений (**X** и **Y** — вещественные параметры). Используя четыре вызова этого метода, найти минимальное и максимальное из заданных с клавиатуры вещественных чисел **A, B, C, D**.
4. Описать метод **TimeToHMS(T, H, M, S)**, возвращающий значение типа **void**, определяющий по времени **T** (в секундах) содержащееся в нем количество часов **H**, минут **M** и секунд **S** (**T, H, M** и **S** — параметры целого типа). Используя метод, найти количество часов, минут и секунд для введенного пользователем с клавиатуры отрезка времени.