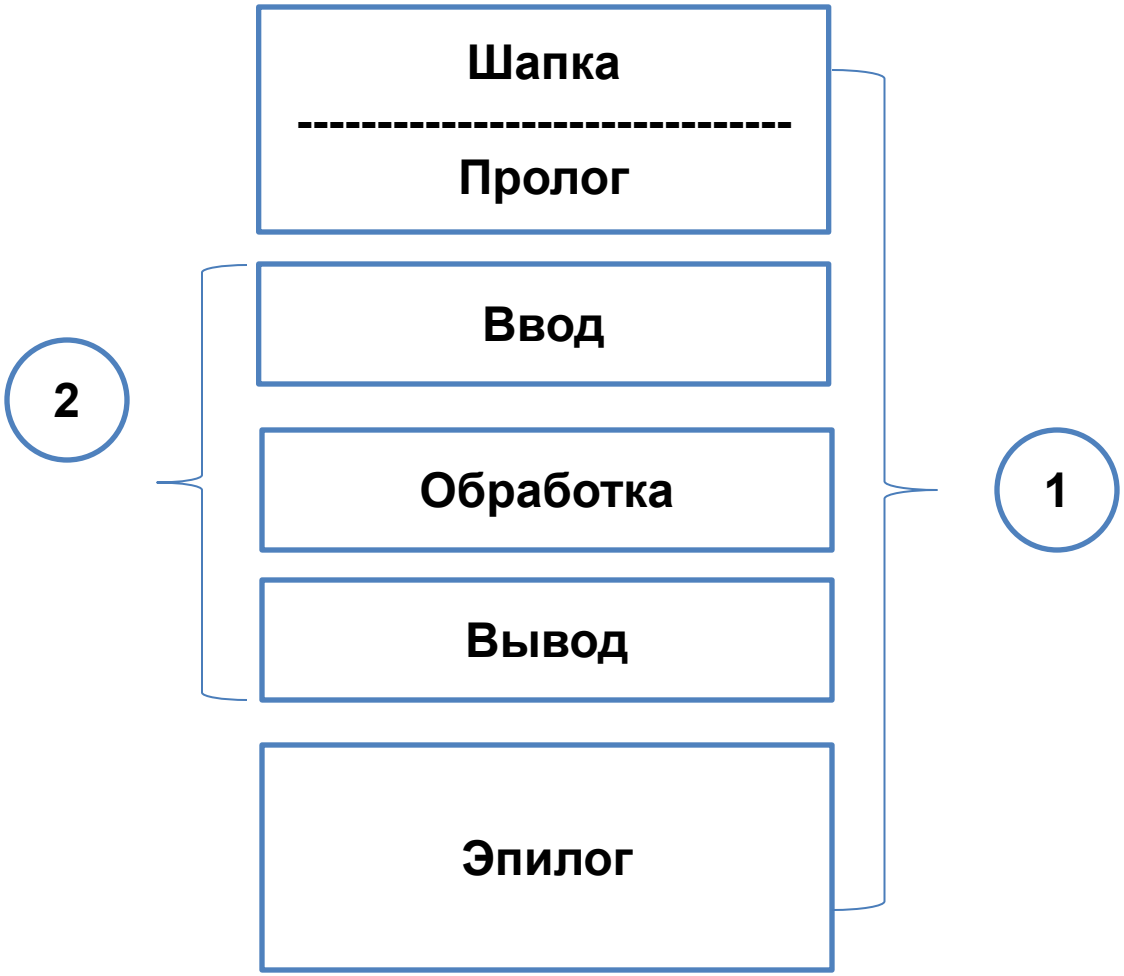
Дисциплина «Программирование» Отделение программной инженерии
В.В. Подбельский, О.В. Максименкова

Модуль 1, Практическое занятие 2

Переменные арифметических типов, тернарная операция, приведение типов

Схема создания программы



Задача 1

Написать метод для вычисления приближенного значения n-го члена ряда Фибоначчи по формуле Бине:

$$U_n = \frac{b^n - (-b^{-n})}{2b - 1}$$

$$b = \frac{1 + \sqrt{5}}{2}$$

В основной программе, вводя с помощью метода int.TryParse() значения n, вычислять и выводить приближенное вещественное значение n-го члена ряда Фибоначчи. Окончание работы программы – ввод нулевого или отрицательного значения n.

Задача 1

Всегда положительное число. Используем беззнаковый целый

1

Пролог

```
using System;
class Program {
   public static void Main() {
      uint n;          // Номер члена ряда
      double b,        // Вспомогательная переменная
             un;       // Оценка по формуле Бине
      int res;         // Целочисленное значение члена
      string line;     // Строка для приёма данных
```

2

```
 Console.WriteLine("Для выхода из программы нажмите ENTER.");
      Console.ReadLine();
   } //Конец определения метода Main()
} //Конец объявления класса Program
```

Эпилог

1

Задача 1. Реализация без метода

```
Console.Write("Введите номер члена ряда: ");
line = Console.ReadLine();
if (!uint.TryParse(line, out n))   {
    Console.WriteLine("Ошибка ввода!");
    return;   }
```

Ввод

```
b = (1 + Math.Sqrt(5)) / 2;
un = (Math.Pow(b, n) - Math.Pow(-b, -n)) / (2*b-1);
res = (int)(un + 0.5);
```

Обработка

```
Console.WriteLine("un = " + un + ", число Фибоначчи: " + res);
```

Вывод

double Math.Pow(double <параметр_1>, double <параметр_2>)

Возвращает значение типа double равное результату возведения <параметр_1> в степень <параметр_2>

http://msdn.microsoft.com/en-us/library/system.math.pow(v=vs.85).aspx

double Math.Sqrt(double <параметр>)

Возвращает значение типа double, равное корню квадратному из <параметр>

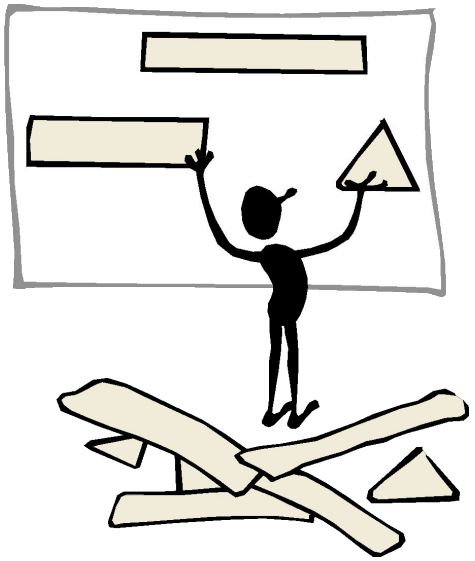
http://msdn.microsoft.com/en-us/library/system.math.sqrt(v=vs.85).aspx

Задача 1. Поэтапно модифицируем код

Этап 1. Выделение метода Un()

- Какой код будет перемещён в метод Un()?
- Какой тип возвращает метод Un()?
- Как изменится код основной программы?

Этап 2. Добавление цикла повторения решения



Задача 1. Этап 1. Выделение метода

```
using System;
class Program {
    public static int Un(uint n) {
        double b = (1 + Math.Sqrt(5)) / 2;
        double un = (Math.Pow(b, n) - Math.Pow(-b, -n)) / (2 * b - 1);
        return (int)(un + 0.5);
    } // Конец определения метода Un()
    public static void Main() {
        uint n;          // Номер члена ряда
        int res;         // Целочисленное значение члена
        string line;    // Строка для приёма данных
        do  {  Console.Write("Введите номер члена ряда: ");
               line = Console.ReadLine();
        }   while (!uint.TryParse(line, out n));
        res = Un(n);       // Вызов метода
        Console.WriteLine("число Фибоначчи: " + res);
        Console.WriteLine("Для выхода из программы нажмите ENTER.");
        Console.ReadLine();
    } //Конец определения метода Main()
} //Конец объявления класса Program
```

Задача 1. Этап 2. Добавление цикла повторения решения

```
//Нажатая пользователем клавиша
ConsoleKeyInfo клавиша;
do   {      // цикл для повторения решения задачи
    /* …. */
    /* код решения задачи*/
    /* …. */
} while (/* условие */);
```

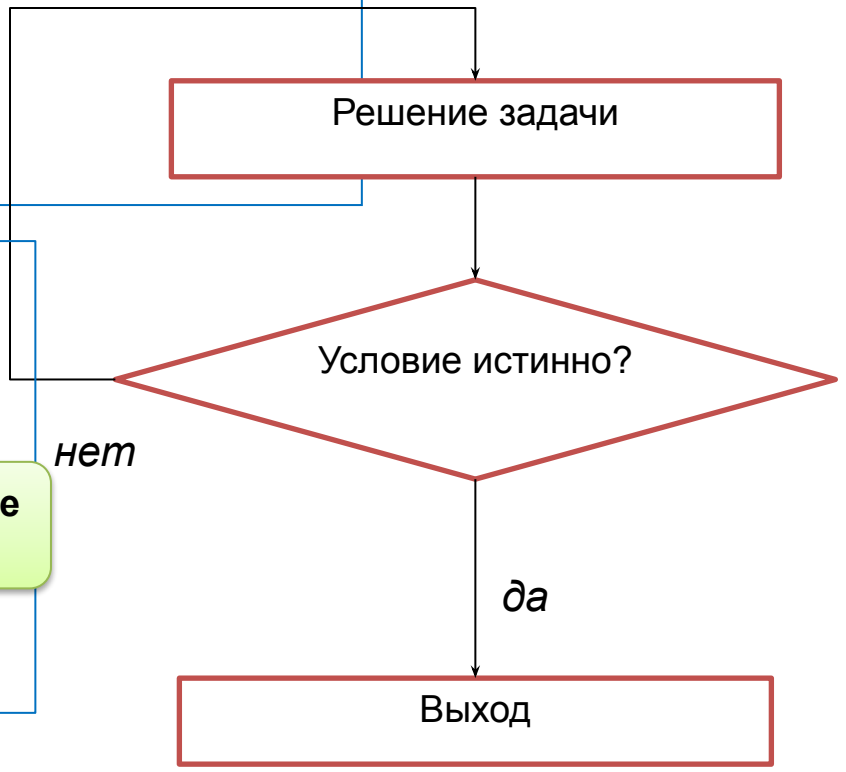
```
do {
        Console.Write("Введите номер члена ряда: ");
        line = Console.ReadLine();
        uint.TryParse(line, out n);

        if (n <= 0) break;

        res = Un(n);
        Console.WriteLine("число Фибоначчи: " + res);
} while (true);
```

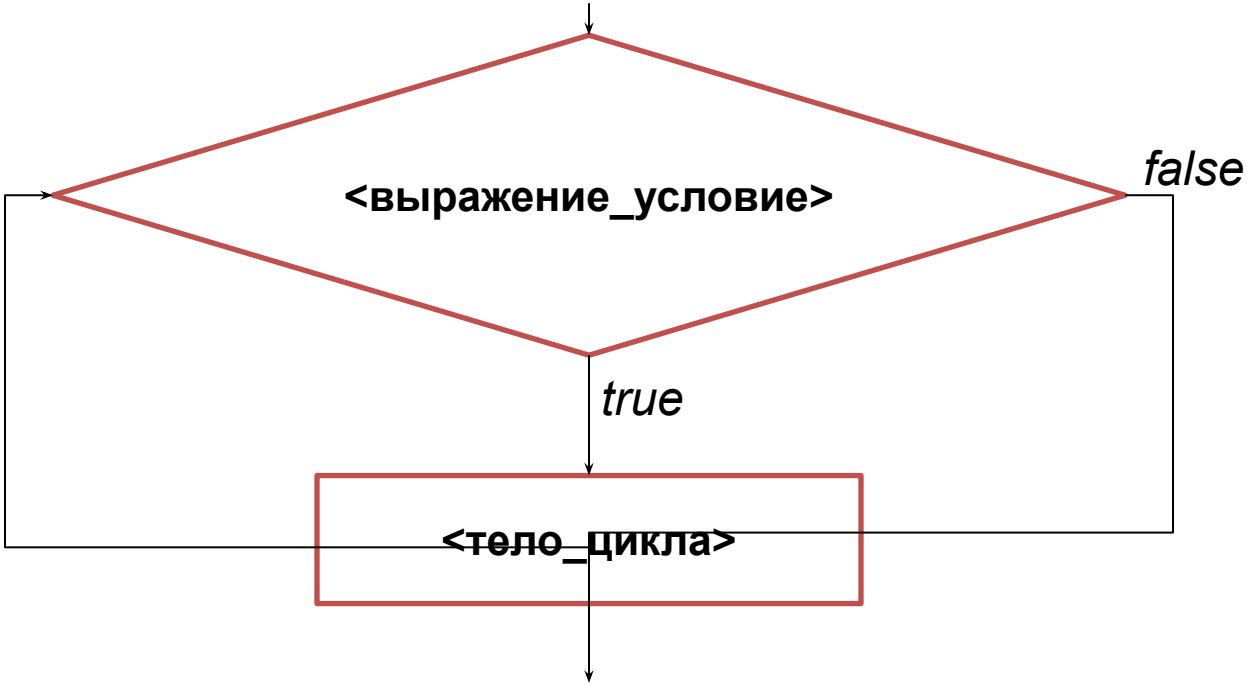
Прерывание цикла

Бесконечный цикл



Задание к задаче 1

#TODO: Организуйте повторение решения в задаче 1 с использованием цикла с предусловием



Например,

```
int i = 5;
while (i > 1) {
   i--;
}
```

Условие	i
5 > 1 == true	5
4 > 1 == true	4
3 > 1 == true	3
2 > 1 == true	2
1 > 1 == false	1

Задача 2. «Шапка» кода программы

```
/*
  Дисциплина: "Программирование"
  Группа:     101ПИ/1
  Студент:    Иванов Иван Иванович
  Задача:     Написать метод для вычисления площади и длины
    окружности, радиус которой задает вещественный
    параметр.
   В основной программе, вводя значения радиуса, с помощью
    обращения к методу, вычислять площадь и длину
    окружности. При вводе применять метод double.TryParse() и
    проверять корректность введенного значения. При выводе
    использовать форматную строку метода WriteLine().
    Окончание работы программы – ввод нулевого или
    отрицательного значения радиуса.
   Дата:       2015.09.17
*/
```

Задача 2. Выделяем этапы

Этап 1. Определение переменных, необходимых для работы основной программы и организация вывода

- Какие нужны переменные?
- Какие у них типы данных?

Этап 2. Получение от пользователя значения радиуса r с применением метода TryParse()

Этап 3. Реализация метода вычисления площади и длины окружности

- Какого типа значение возвращает метод?
- Как вернуть получить из метода значения и длины окружности и площади одновременно?

Этап 4. Вызов метода из основной программы

Этап 5. Организация форматированного вывода

Этап 6. Организация повторения решения

Задача 2. Этап 1. Объявление переменных

```
using System;
class MyExample  {
    public static void Main()   {
        double r, //радиус
         s, // площадь круга
         c;  // длина окружности
        string str;       // Рабочая строка для ввода и вывода данных
```

Проло

Все значения могут быть вещественными числами. Используем тип double

1

2

```
        Console.WriteLine(str);        // выводим результат
        Console.WriteLine("Для выхода из программы нажмите ENTER.");
        Console.ReadLine();
    } //Конец определения метода
} //Конец объявления класса
```

Эпило

г

1

Задача 2. Этап 2. Получение значения радиуса

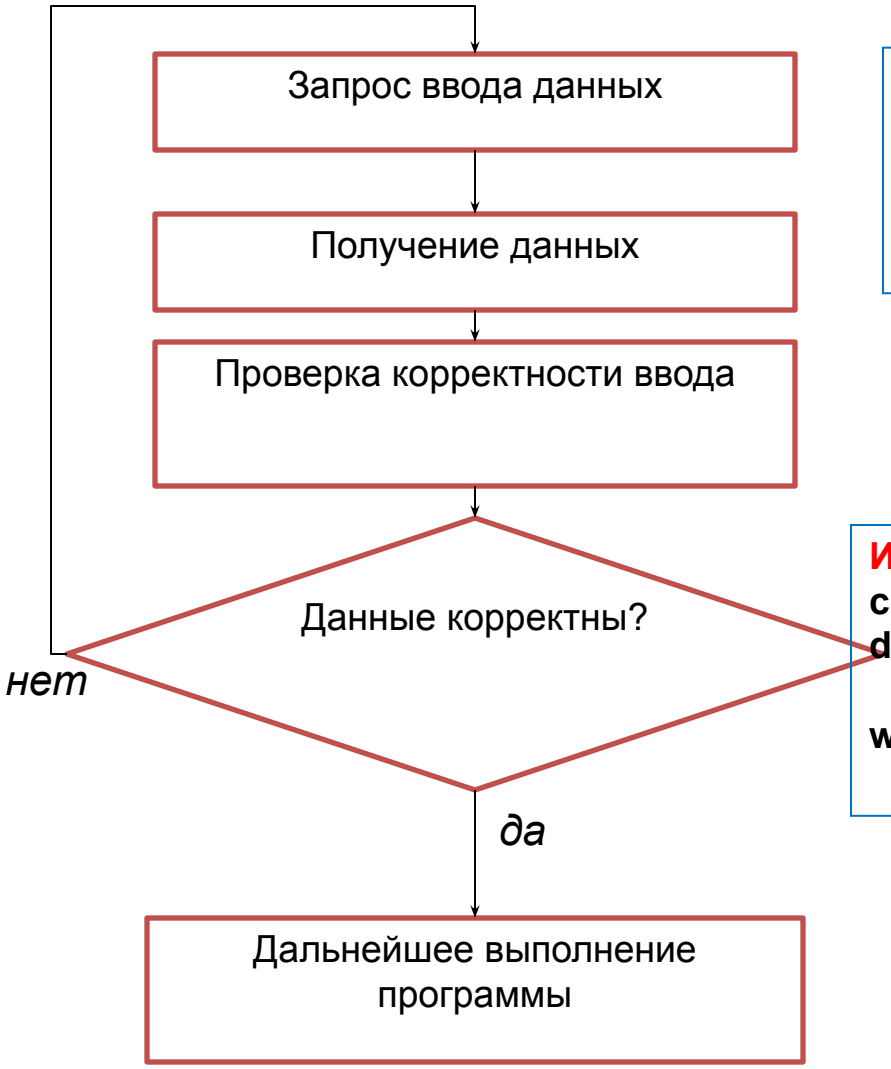
Шаг 1. Получить значение радиуса МОЖНО БЫЛО БЫ ТАК:

~~Console.Write("Введите радиус: ");~~
~~str = Console.ReadLine(); //Читаем символьную строку~~
~~double.TryParse(str, out r); //Преобразуем строку в число~~

Шаг 2. Модифицируем ПРЕДЛОЖЕННЫЙ код, чтобы проверялась корректность введённого значения

```
do {
        Console.Write("Введите радиус: ");
        str = Console.ReadLine();       //Читаем символьную строку
        double.TryParse(str, out r);
} while (!double.TryParse(str, out r)); //Преобразуем строку в число
```

Проверка корректности ввода данных



Например,

```
double x;
do
    Console.Write("Введите число: ");
while (!double.TryParse(Console.ReadLine(), out x));
```

Или, когда недопустим КОНКРЕТНЫЙ символ ‘q’:

```
char ch;
do
     Console.Write("Введите символ:  ");
while (!char.TryParse(Console.ReadLine(), out ch)
                    || ch=='q');
```

Методы обработки ввода (повторение 1-го семинара)

```
a = int.Parse(str);
```

<тип> <тип>.Parse(<строка>)

преобразует строковое представление числа в эквивалентное ему число типа <тип>. В случае, если строковое представление не может быть преобразовано к соответствующему типу, возникает исключение. ОБРАБОТКУ ИСКЛЮЧЕНИЙ РАССМОТРИМ ЧУТЬ ПОЗЖЕ!

```
int.TryParse(str, out b);
```

bool <тип>.TryParse(string <строка>, out <тип> <параметр>)

Метод преобразует строковое представление числа (первый параметр) в эквивалентное ему числовое значение типа <тип> и присваивает это значение переменной, использованной в качестве аргумента, заменившего второй параметр метода. Метод возвращает значение true, если преобразование выполнено успешно и false – в противном случае.

Задача 2. Этап 3. Реализация метода

```
using System;
class MyExample  {
   public static double Square(double r, out double len) {
      len = 2 * Math.PI * r;
      return Math.PI * r * r;
   } //Конец определения Square()
   public static void Main()   {
      double  r, //радиус
                 s, // площадь круга
                 c;  // длина окружности
      string str;       // Рабочая строка для ввода и вывода данных
      do {
         Console.Write("Введите радиус: ");
         str = Console.ReadLine();         //Читаем символьную строку
      } while (!double.TryParse(str, out r)); // Преобразуем строку в число

      // Вызов метода и строку с результатом ОФОРМИТЕ САМОСТОЯТЕЛЬНО!
      Console.WriteLine(str);         // выводим строку с результатом
      Console.WriteLine("Для выхода из программы нажмите ENTER.");
      Console.ReadLine();
   } //Конец определения Main()
} //Конец объявления MyExample
```

Задание к задаче 2

#TODO: Самостоятельно выполните эпап 5. Организуйте вывод информации о площади и длине окружности, используя форматную строку методов Console.Write() и Console.WriteLine().

#TODO: Самостоятельно выполните этап 6.

Структура поля подстановки:

{N[,W][:S[R]]}, где

N – номер аргумента;

W – ширина поля;

S – спецификатор формата;

R – спецификатор точности.

Спецификатор S:

C или c – валютный

D или d – целочисленный

E или e – экспоненциальный

F или f – с фиксированной точкой

G или g – «короткий» из E или F

X или x – шестнадцатеричный

Статья «Строки стандартных числовых форматов на msdn.microsoft.com - http://msdn.microsoft.com/ru-ru/library/dwhawy9k.aspx
Статья «Составное форматирование» на msdn.microsoft.com - http://msdn.microsoft.com/ru-ru/library/txafckwd.aspx

Форматирование строк при выводе

```
int i = 3;
double d = 1 / (double)i;
```

```
Console.WriteLine("{0} - {1}", i, d);
```

#0 #1

```
3 - 0,333333333333333
```

```
Console.WriteLine("{0} - {0}", i, d);
```

```
3 - 3
```

```
Console.WriteLine("{0} - {1:f5}", i, d);
```

```
3 - 0,33333
```

```
Console.WriteLine("{1} - {2}", i, d);
```

Исключение!!!

Задача 3

/* Задача на применение тернарной операции.
(Не применять оператор if.)
Написать метод, так обменивающий значения трех переменных x, y, z, чтобы выполнилось требование: x <= y <= z.

В основной программе, вводить значения трех переменных и упорядочивать их с помощью обращения к написанному методу.

Для выхода из программы вводите ESC, для повторения решения – любую другую клавишу.
*/

Задача 3

```
using System;
class Program {
    public static void Main()    {
        int x=0, y=0, z=0;
        string str;    // Строка для приёма данных
```

#TODO: Самостоятельно реализуйте получение от пользователя значений переменных x, y и z.

```
        Console.WriteLine("x = " + x);
        Console.WriteLine("y = " + y);
        Console.WriteLine("z = " + z);
} //Конец метода Main()
} //Конец объявления класса Program
```

Задача 3

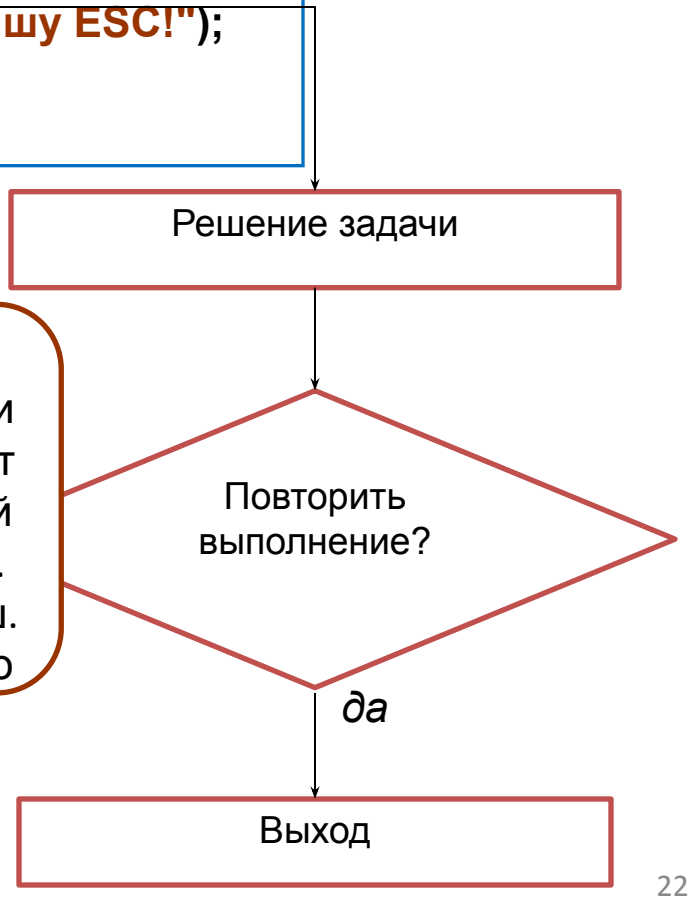
```
public static void ValuesSort(ref int x, ref int y, ref int z) {
        // Вспомогательные переменные
        int a1 = 0, a2 = 0, a3 = 0;
        a1 = x < y ? (z < x ? z : x) : (y < z ? y : z);
        a3 = x > y ? (z > x ? z : x) : (y > z ? y : z);
        // Самостоятельно определите значение a2 !!!!
        x = a1; y = a2; z = a3;
    }
```

#TODO: Самостоятельно реализуйте вызов метода из основной программы.

#TODO: .Дополните основную программу циклом повторения решения, согласно условиям задачи. Для справки используйте следующий слайд.

Задача 3. Повторение решения

```
//Нажатая пользователем клавиша
ConsoleKeyInfo клавиша;
do  {      // цикл для повторения решения задачи
    /* …. */
    /* код решения задачи*/
    /* …. */
    Console.WriteLine("Для выхода нажмите клавишу ESC!");
    клавиша = Console.ReadKey(true);
} while (клавиша.Key != ConsoleKey.Escape);
```



ConsoleKeyInfo Console.ReadKey(bool intercept) получает код нажатой пользователем клавиши (символьной или функциональной). Метод возвращает ссылку на объект типа **ConsoleKeyInfo**. Код нажатой клавиши консоли - **Console.ReadKey(true).Key**. Перечисление **ConsoleKey** задает коды клавиш. Параметр **intercept** - признак визуализации вводимого символа.

Задача 4

/* Задача 5.
Написать метод с целочисленным параметром, определяющий является ли значение аргумента кодом цифры, кодом буквы русского алфавита (прописной либо строчной), или ни тем и ни другим. В основной программе, вводя целые числа, выводить сообщения о них: «Это цифра!», «Это буква!», «Это ни буква, ни цифра!». Для выхода из программы – ESC, для повторения решения - любой символ. При анализе цифрового кода использовать тернарную операцию. Строку сообщения может возвращать метод, либо метод возвращает признак, а строку формирует основная программа
*/

Задача 4

```
using System;
class Program   {
   public static void Main()   {
      uint code;
      string str;              // Строка для приёма данных
      string report;           // Строка с заключением о коде
      uint код_A = (uint)'A',  // Числовое значение кода буквы A
           код_а = (uint)'а',
           код_я = (uint)'я',
           код_Я = (uint)'Я‘,
           код_0 = (uint)‘0‘;   // Числовое значение кода цифры 0
```

Пролог

2

```
      Console.WriteLine("Для выхода из программы нажмите ENTER.");
      Console.ReadLine();
   } //Конец определения метода Main()
} //Конец объявления класса Program
```

Эпилог

Задача 4

```
Console.WriteLine("Коды граничных символов:");
Console.WriteLine("Код А = " + код_А + "; Код Я = " + код_Я +
        "; Код а = " + код_а + "; Код я = "+ код_я +
        "; Код нуля = " + код_0);
Console.Write("Введите значение code: ");
 str = Console.ReadLine();
uint.TryParse(str, out code);
```

Ввод

```
report = code <= '9' && code >= '0' ? "Это цифра: " + (char)code
      : code <= 'Я' && code >= 'А' ? "Это прописная буква: " + (char)code
       : code <= 'я' && code >= 'а' ? "Это строчная буква: " + (char)code
       : "Неизвестный символ!";
Console.WriteLine(report);
```

Обработка – тело метода

Задача 5

```
/*
Выделение отдельных цифр натурального числа

Задача.
Ввести трехзначное натуральное число и
напечатать его цифры "столбиком".
*/
```

Задача 5

```
using System;
class Program    {
   public static void Main()    {
      uint numb,       // Введенное число
           figure,     // Очередная цифра
           temp;       // рабочая переменная
      string str;      // Строка для приёма данных
```

Пролог

2

```
Console.WriteLine("Для выхода из программы нажмите ENTER.");
      Console.ReadLine();
   } //Конец определения метода Main()
} //Конец объявления класса Program
```

Эпилог

Задача 5

Ввод

```
Console.Write("Введите целое от 100 до 999: ");
str = Console.ReadLine();
uint.TryParse(str, out numb);     // НУЖНА проверка смысла
```

Обработка

```
figure = numb / 100;              // первая цифра
Console.WriteLine(figure);
temp = numb - figure * 100;
figure = temp / 10;               // вторая цифра
Console.WriteLine(figure);
figure = temp - figure * 10;      // третья цифра
Console.WriteLine(figure);
```

```
using System;
namespace _2_5  {
   class Program    {
      static void Numerals(uint  number, out uint f1, out uint f2, out uint f3) {
         f1 = number / 100;          // первая цифра
         uint  temp = number - f1 * 100;
         f2 = temp / 10;           // вторая цифра
         f3 = temp - f2 * 10;       // третья цифра
         }
      static void Main( )       {
         uint numb = 759;
         do       Console.Write("Введите целое от 100 до 999: ");
         while (!uint.TryParse(Console.ReadLine(), out numb)|| numb < 100 || numb >999);
         uint a, b, c;
         Numerals(numb, out a, out b, out c);
         Console.WriteLine(a);
         Console.WriteLine(b);
         Console.WriteLine(c);
      }
   }
}
```

Добавить повторение решения задачи

Задача 6

```
/*
Задан круг с центром в начале координат и радиусом R=10.
Ввести  координаты точки и вывести сообщение: «Точка внутри круга!» или «Точка вне круга!».

Предусмотреть проверку входных данных и цикл повторения решений.
*/
```

Задача 6

```
using System;
class Program    {
static void Main()       {
    const double R=10;
    double x, y;
    Console.Write("x = ");
    double.TryParse(Console.ReadLine(), out x) ;
    Console.Write("y = ");
    double.TryParse(Console.ReadLine(), out y) ;
    string report = "Точка ";
```

Пролог

Ввод

```
report += x*x + y*y > R*R ? "вне круга!" : "внутри круга!";
```

Обработка

```
    Console.WriteLine(report);
    Console.WriteLine("Для выхода из программы нажмите ENTER.");
    Console.ReadLine();
    }
}
```

Эпилог

Задача 7

```
//Вычисление значения логической функции !(X&Y|Z)
using System;
class Program    {
   public static void Main()    {
      int tmp;        //рабочая переменная
            bool x, y, z;  //аргументы логической функции
            string str;    //строка для приема данных
```

Пролог

2

```
Console.WriteLine("Для выхода из программы нажмите ENTER.");
 Console.ReadLine();
   } //Конец определения метода Main()
} //Конец объявления класса Program
```

Эпилог

Задача 7

```
// Пусть положительное значение изображает TRUE
Console.Write("Введите значение x: ");
str = Console.ReadLine();
int.TryParse(str, out tmp);
```

Ввод

```
x = tmp > 0 ? true : false;
```

Обработка

Повторяем эти действия для переменных y и z, затем выводим на экран результат вычисления:

```
Console.WriteLine("!(X&&Y||Z) = " + !(x && y || z));
```

Вывод

Решите самостоятельно. Напишите методы и обращения к ним

Ввести значение x и вывести значение полинома:
$F(x) = 12x^4 + 9x^3 - 3x^2 + 2x - 4$. Не применять возведение в степень.
Использовать минимальное количество операций умножения.

Ввести натуральное трехзначное число P. Найти наибольшее целое число, которое можно получить, переставляя цифры числа P.

Введя значения коэффициентов A, B, C, вычислить корни квадратного уравнения. Учесть (как хотите) возможность появления комплексных корней. Оператор if не применять.

Ввести четырехзначное натуральное число и напечатать его цифры в обратном порядке.

Введя значения напряжения U и сопротивления R, вычислить силу тока $I = U/R$ и потребляемую мощность $P = U^2/R$ электрической цепи.