Основы .NET разработки

Тема 1. Классификация языков программирования. Структура программы на C#. Первая программа

Классификация языков программирования

- Язык программирования — формальный язык, предназначенный для записи компьютерных программ. Язык программирования определяет набор лексических, синтаксических и семантических правил, определяющих внешний вид программы и действия, которые выполнит исполнитель (обычно — ЭВМ) под её управлением

Машинный код

Низкоуровневые

Высокоуровневые

Сверх

Классификация языков программирования

- Уровень языка зависит не от его «крутости», а от того на сколько близко его команды похожи на команды процессора.

- Самые процессорные языки это машинный код, затем низкий уровень, высокий уровень, сверхвысокий уровень и обычный человеческий язык

Классификация языков программирования

машинный код

- **Машинный код** — это команды понятные компьютеру, состоящие из нулей и единиц

Классификация языков программирования

Низкоуровневые

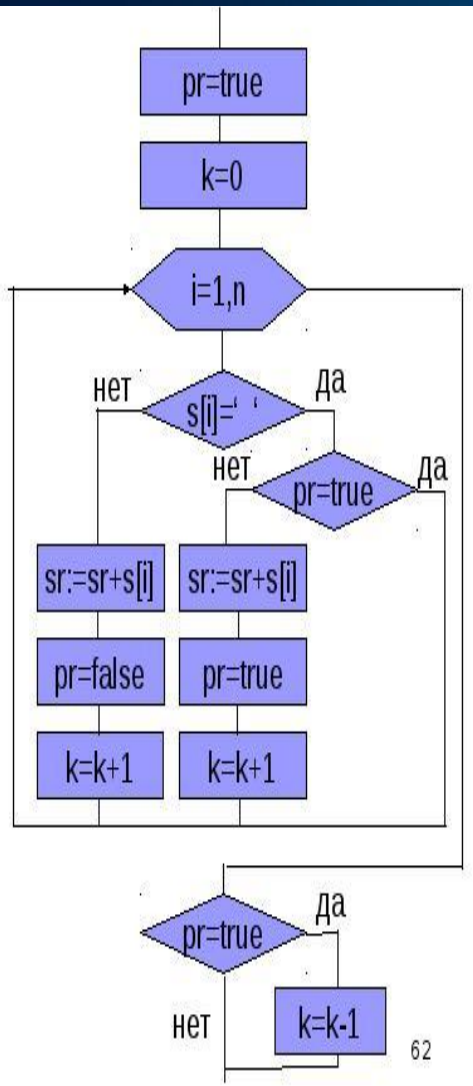
- **Низкоуровневые** языки программирования, близкие к программированию непосредственно в машинных кодах используемого реального или виртуального процессора. Для обозначения машинных команд обычно применяется мнемоническое обозначение. Это позволяет запоминать команды не в виде последовательности двоичных нулей и единиц, а в виде осмысленных сокращений слов человеческого языка. Примеры языков:

Ассемблер, Forth, C

Классификация языков программирования

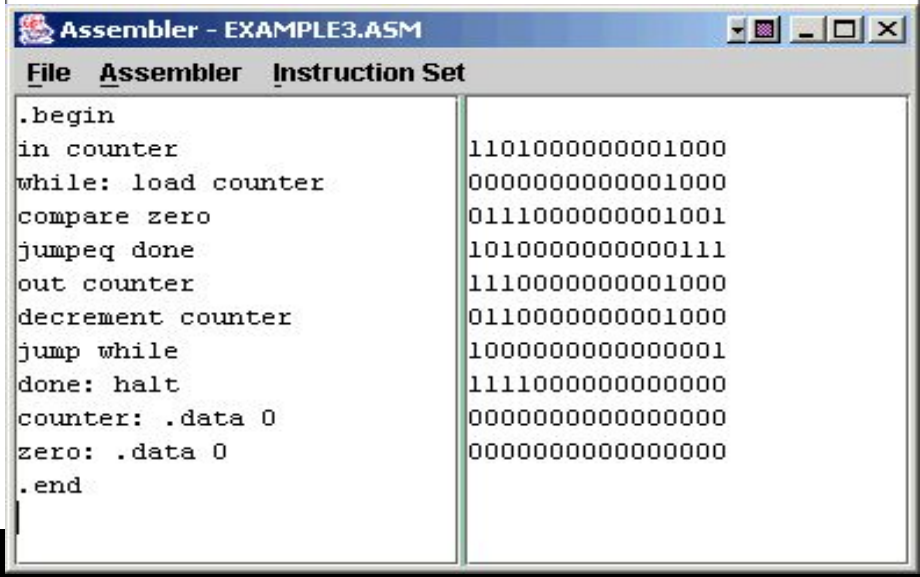
Низкоуровневые

```
        mov     DL,0
        jcxz    prod3
        mov     BX,1
        cld
cycl1:  lodsb
        cmp     AL,' '
        je      prod1
        mov     BX,0
        inc     DL
        stosb
        jmp     prod2
prod1:  cmp     BX,1
        je      prod2
        mov     BX,1
        inc     DL
        stosb
prod2:  loop    cycl1
```



Синтаксис языка Ассемблера

Пример программы в машинном коде



Классификация языков программирования

Высокоуровневые

- Высокоуровневый — язык программирования, разработанный для быстроты и удобства использования программистом. Основная черта высокоуровневых языков — это абстракция, то есть введение смысловых конструкций, кратко описывающих такие структуры данных и операции над ними, описания которых на машинном коде (или другом низкоуровневом языке программирования) очень длинны и сложны для понимания.

C++, Pascal, Java, C#, PHP, ObjectC, Delphi и др.

Классификация языков программирования

Высокоуровневые

```
.model small
   .stack 256
   .data
   filda   dw  1
   fildb   dd  1.5
.code
_main  proc
   PUBLIC    main
   EXTRN   @cproc$qif:proc
   ;EXTRN    cproc:proc
   mov   ax,@data
   mov   ds,ax
   push  fildb
   push  filda
   call    @cproc$qif
   ;call    cproc
   add   sp,6
.exit   0
_main  endp
   end _main
```

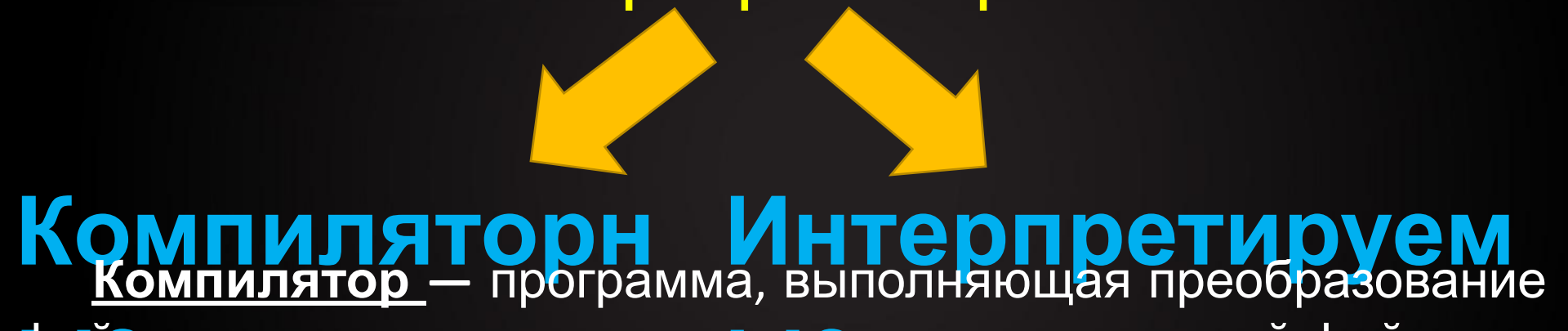
```
#include <iostream.h>
void   cproc(int x,float y)
{
cout << "Rez " << x << " “;
cout << y << "\n";
}
```

Классификация языков программирования

Сверх высокоуровневый

- Сверх высокоуровневый язык программирования (англ. very high-level programming language, VHLL) — язык программирования с очень высоким уровнем абстракции. В отличие от языков программирования высокого уровня, где описывается принцип «как нужно сделать», в сверх высокоуровневых языках программирования описывается лишь принцип «что нужно сделать».. **Python, Ruby, Haskel, Perl, мини язык AWK**

Классификация языков программирования Высокого уровня



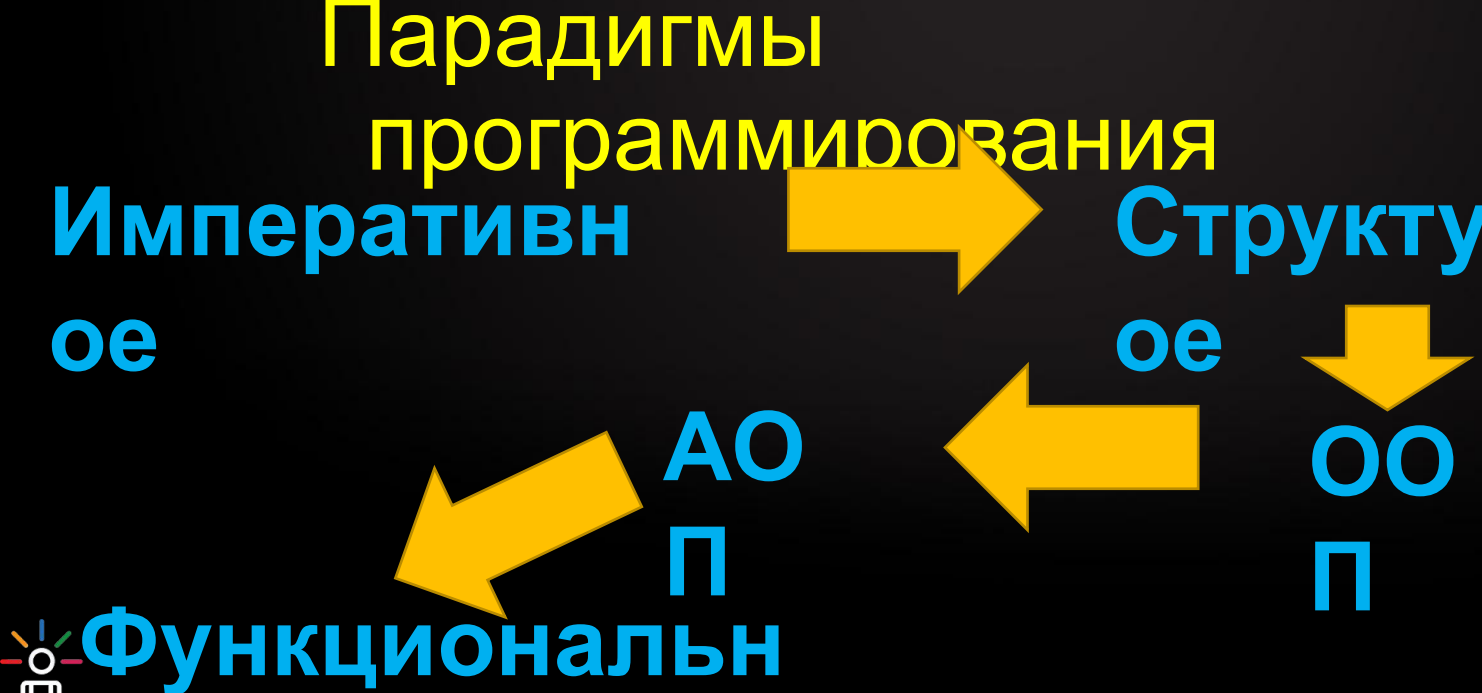
Компилятор — программа, выполняющая преобразование файла с исходным кодом программы в исполняемый файл

Компиляция — преобразование программы, составленной на исходном языке высокого уровня в эквивалентную программу на низкоуровневом языке (машинном коде).

Интерпретатор — это программа, которая получает исходную программу и по мере распознавания конструкций входного языка реализует действия, описываемые этим и конструкциями

Парадигмы программирования

Парадигма программирования — это совокупность идей и понятий, определяющих стиль написания компьютерных программ (подход к программированию)



Императивное программирование процедурное

Программирование, при котором последовательно выполняемые операторы можно собрать в подпрограммы, то есть более крупные целостные единицы кода, с помощью механизмов самого языка. Выполнение программы сводится к последовательному выполнению операторов с целью преобразования исходного состояния памяти, то есть значений исходных данных, в заключительное, то есть в результаты.

Структурное программирование

Парадигма программирования, в основе которой лежит представление программы в виде иерархической структуры блоков.

В соответствии с парадигмой, любая программа строится без использования оператора goto из трёх базовых управляющих структур: последовательность, ветвление, цикл

Принципы структурного программирования

- Следует отказаться от использования goto
- Три базовые управляющие инструкции: последовательность, ветвление, цикл
- Вложенность управляемых конструкций
- Повторы оформлять в процедуры и функции
- Каждую логически законченную группу инструкций следует оформить как блок.
- Конструкции должны иметь один вход и один выход
- Подход «сверху вниз» «снизу вверх»

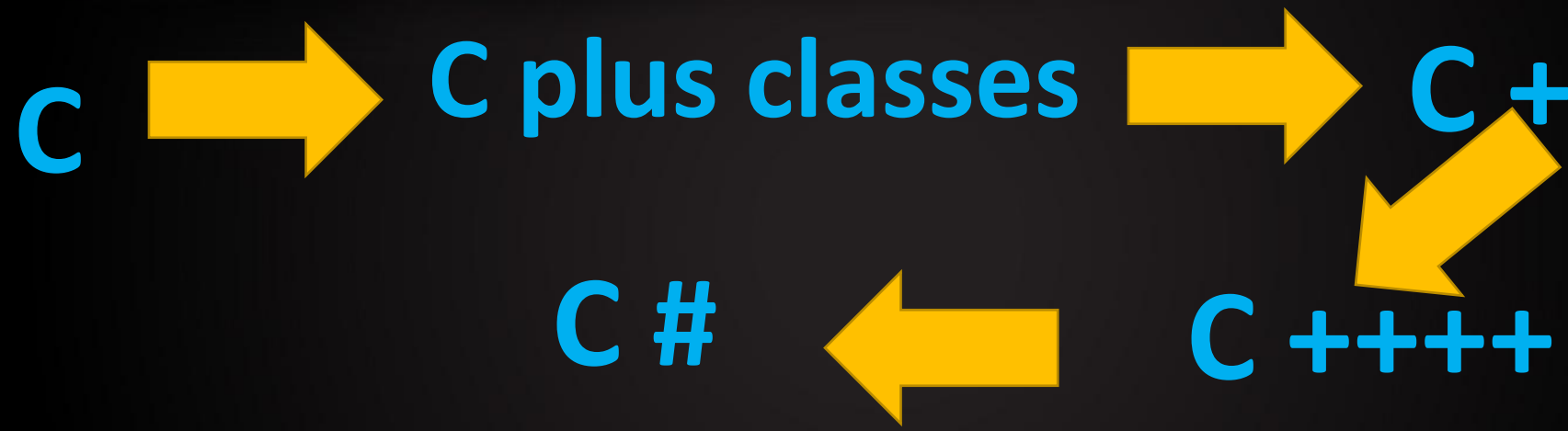
ООП программирование

Объектно-ориентированное программирование (ООП) — парадигма (методология) программирования, основанная на представлении программы в виде совокупности объектов, каждый из которых является экземпляром определённого класса, а классы образуют иерархию наследования.

Базовые принципы ООП

- Абстрагирование
- Инкапсуляция
- Полиморфизм
- Наследование
- Объект
- Класс

C#



Перед тем как изучать любой язык будь то русский, белорусский, английский и т.п. нужно изучить его алфавит. В языке C# также есть свой алфавит.

Алфавит c#

- A..Z a..z (_) как буква РУВАЕТСК
- Все цифры (0..9)
- ! “ @ # % ^ & * ( ) + - / \ | { } ~ [ ] : ; < > = ,
- //однострочные /*многострочные*/

Лексемы C#

- Идентификаторы
- Ключевые слова
- Знаки и символы операций
- Литералы
- Разделители

Идентификаторы

- **Идентификаторы** — имя чего-либо, состоящее из последовательности символов.
- В языке C# идентификаторами являются:
- типы данных, имена переменных, функций, классов, интерфейсов, делегатов;
- Могут состоять из букв (A..Z a..z) цифры (0..9) и _
- **Нельзя начинать с цифры, прописные и строчные буквы это разные символы недолжны совпадать с ключевыми (зарезервированными) словами.**

Ключевые слова

abstract	as	base	null	operator	break
byte	case	catch	override	private	char
checked	class	const	public	ref	continue
decimal	default	delegate	sbyte	short	do
double	else	enum	stackalloc	string	event
explicit	extern	false	switch	throw	finally
fixed	float	for	try	uinit	foreach
goto	if	implicit	unchecked	ushort	in
int	interface	internal	void	while	is
lock	long	namespace	object	new	return
params	sealed	this	unsafe	out	sizeof
readonly	static	typeof	volatile	protected	struct
true	ulong	using	bool	virtual	

Знаки и символы операций
Литералы

+ - ++ -- * / = == != < <= > >= += -= *= /= % ||

- Литералами называют представление значения некоторого типа данных;
- Разделители (пробелы, табуляторы переход на новую строку).

Структура программы на C#

- Подключение/объявление пространства имен (своего рода контейнера);
- объявление класса
- методы класса (подпрограммы), как минимум метод Main;
- операторы и выражения;

в каждом из этих блоков могут присутствовать комментарии — участок кода, невосприимчивый компилятором.

Структура программы на C#

```
1   //подключение пространства имен
2   using System;
3   using System.Collections.Generic;
4   using System.Linq;
5   using System.Text;
6   //объявление пространвнсва имен
7   namespace ConsoleApp1
8   {
9       //оъявление класса
10      class Program
11      {
12          //методы класса
13          static void Main(string[] args)
14          {
15              //тело метода операторы, выражения
16          }
17      }
18  }
```

Практическая часть

№	Задание
1	Задание написать программу вывода строчки “Hello world” на экран. сделать это при помощи обычного текстового редактора и компилятора C#
2	Задание написать программу вывода строчки “Hello world” на экран.сделать это в любой из IDE (среды разработки), например VisualStudio

Практическая часть

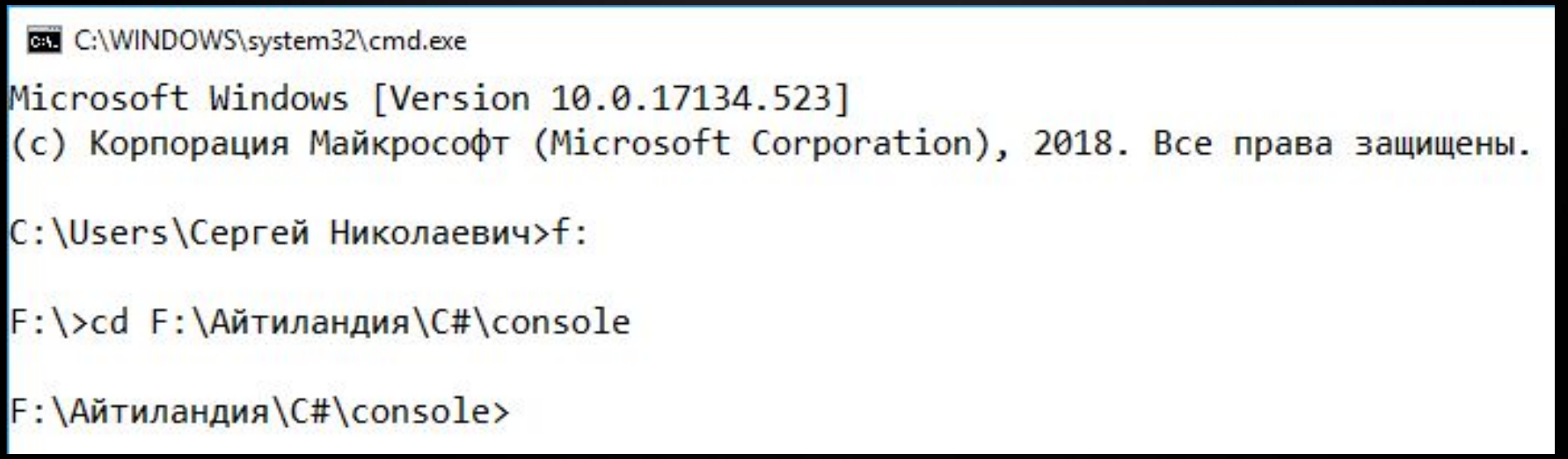
решение1

Набираем текст в любом текстовом редакторе и сохраняем его с расширением .cs под любым именем (programm.cs) на любой из дисков в любую папку например F:\Айтиланд

```
1  using System;
2
3  namespace HelloWorld
4  {
5      class Program
6      {
7          static void Main(string[] args)
8          {
9              Console.WriteLine("Hello World!!!");
10             Console.ReadLine();
11         }
12     }
13 }
```

Практическая часть решение1

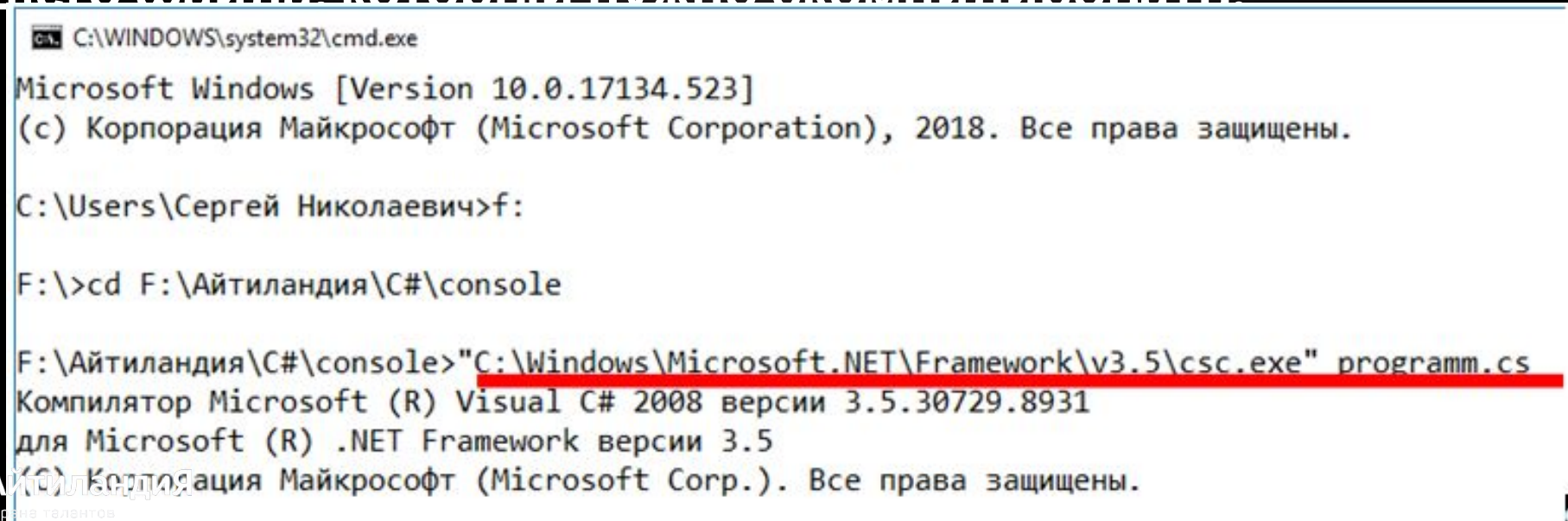
Далее заходим в командную строку (win+R→cmd→enter) и переходим в папку где находится наш документ



Практическая часть

решение1

Далее прописываем путь где может находится компилятор для этого в поиске на диске С напишите csc.exe это и есть компилятор и компьютер вам выдаст путь к нему обычно это тут (C:\Windows\Microsoft.NET\Framework\v3.5) впишите путь к компилятору в кавычки и через пробел пропишите имя вашего файла, который нужно скомпилировать.



Практическая часть

решение1

После компиляции у вас появится второй файл с таким же названием но с расширением .exe

```
Содержимое папки F:\Айтиландия\C#\console

21.01.2019  14:40    <DIR>          .
21.01.2019  14:40    <DIR>          ..
20.01.2019  13:23               293 programm.cs
21.01.2019  14:40             3 584 programm.exe
               2 файлов          3 877 байт
               2 папок  76 343 365 632 байт свободно
```

И теперь можно запустить эту программ, прописав название документа.exe

```
F:\Айтиландия\C#\console>programm.exe
Hello World!!!
exit
```

Практическая часть

решение2

Загрузить любую IDE (например VisualStudio) создать проект

и написать данный код

```
7           static void Main(string[] args)
8           {
9               Console.WriteLine("Hello World!!!");
10              Console.ReadLine();
11          }
```

Нажать запуск или F5 и посмотреть что получилось

Домашнее задание

№	Задание
1	Установить любую IDE для работы C# предпочтительней VisualStudio
2	Написать программу вывода строчки на экран (меня зовут ФИО. Я начинаю изучать C#)

Спасибо за внимание

АйтиландиЯ
страна талантов