

# С#. Введение.

**Лектор:** Юрий Коноплев

29.07.19

# о курсе и план на неделю

- синтаксис
- ООП в C#
- стандартная библиотека и алгоритмы
- [ASP.NET](#)
- back-end как он есть

# о лекторе

- профиль: мозг-компьютерное взаимодействие
- Perl мой первый серьёзный язык программирования
- запускал стартап TuSion в Нью-Йорке
- работал в DELL, финтех/blockchain компаниях: Aintsys, Waves
- основными языками были C++/Python
- аспирантура Институт Мозга Человека РАН

# КОММУНИКАЦИЯ

- telegram: @bimodaling
- tele-group: [t.me/PIITSchool](https://t.me/PIITSchool) (Phoenix Ingostrah IT School)
- IRC [chat.freenode.net](https://chat.freenode.net) (mIRC, HexChat): #itweeks
- [https://github.com/j0k/it\\_school\\_weeks](https://github.com/j0k/it_school_weeks)

# ССЫЛКИ

- <https://docs.microsoft.com/ru-ru/dotnet/csharp/language-reference/>
- <https://www.learncs.org/>
- <https://www.tutorialsteacher.com/csharp/csharp-tutorials>
- <https://repl.it/languages/csharp>
- [http://www.rosettacode.org/wiki/Rosetta\\_Code](http://www.rosettacode.org/wiki/Rosetta_Code)
- [https://github.com/j0k/it\\_school\\_weeks](https://github.com/j0k/it_school_weeks)

# план на день

- C# и его возможности
- code-style
- конструкции языка: типы переменных, управляющие конструкции
- консольное приложение в Visual Studio
- алгоритм quicksort

# почему C#

- язык общего назначения
- компилируемый
- императивный, ООП и функциональный
- развитая инфраструктура
- CLR - Common Language Runtime
- Anders Hejlsberg (2000)

# задачи на день

- консольное приложение (создание пустого проекта в VS)
- ввод данные и вывод на экран
- запись данных в переменные
- кодирование ДНК
- fortune приложение
- эффект матрицы
- алгоритм quicksort и sort стандартной библиотекой
- определение типов
- CRUD variable application - Create/Read/Update/Delete



# Hello World

```
using System;

namespace ConsoleApp
{
    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            Console.WriteLine("Hello World!");
        }
    }
}
```

# Hello World

```
class Program
{
    0 references
    static void Main(string[] args)
    {
        System.Console.WriteLine("Hello World!");
    }
}
```

# ЗД: CRUD приложение

- консольное приложение
- позволяет создавать, читать, обновлять и удалять переменные
- позволит выводить все переменные по идентификатору
- по аналогии с crudapp будут работать все хранимые сущности: комментарии, посты, записи, личные сообщения

# ЗД: CRUD приложение

```
1 > create int a 10
2 < a=10
3
4 > read int a
5 < 10
6
7 > update int a 20
8 < a=20
9
10 > create string a JustDoIt
11 < a=JustDoIt
12
13 > read a
14 < a:string=JustDoIt
15 < a:int=20
16
17 > delete a
18 < del(a:string)
19 < del(a:int)
20
```

V

# СИНТАКСИС

- переменные
- приведение типов
- if/else/ ?:
- switch
- while, do{}while()
- for, foreach
- IO: WriteLine, ReadLine
- methods

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>

# задача: кодирование ДНК

строка  $S$  - последовательность ДНК

$S = \langle l_1, l_2, l_3, \dots, l_n \rangle$

$l_i \in \{0, 1, 2, 3\}, l_i \in \{'A', 'G', 'T', 'C'\}$

задача: по входной строке вида "021123" вывести

1. последовательность "ATGGTC"

2. цепочку полных названий

"Adenin-Thymine-Guanine-Guanine-Thymine-Cytosine"

# задача: кодирование ДНК

## задача 1

создать переменную в которой храним  
аминокислоту

# переменные

Type	Example
Integral types	sbyte, byte, short, ushort, int, uint, long, ulong, and char
Floating point types	float and double
Decimal types	decimal
Boolean types	true or false values, as assigned
Nullable types	Nullable data types

```
int i, j, k;  
char c, ch;  
float f, salary;  
double d;
```

```
int d = 3, f = 5;  
byte z = 22;  
double pi = 3.14159;  
char x = 'x';
```

- [https://www.tutorialspoint.com/csharp/csharp\\_variables.htm](https://www.tutorialspoint.com/csharp/csharp_variables.htm)



# переменные

```
int myInt = 1;  
float myFloat = 1f;  
bool myBoolean = true;  
string myName = "John";  
char myChar = 'a';  
double myDouble = 1.75;
```

```
var x = 1;  
var y = 2;  
var sum = x + y;
```

- [https://www.learncs.org/en/Variables\\_and\\_Types](https://www.learncs.org/en/Variables_and_Types)

# nullable переменные

```
int? num1 = null;  
int? num2 = 45;  
  
double? num3 = new double?();  
double? num4 = 3.14157;
```

- [https://www.tutorialspoint.com/csharp/csharp\\_nullable.htm](https://www.tutorialspoint.com/csharp/csharp_nullable.htm)

# enum переменные

```
public enum Color { Red, Green, Blue }
```

```
public enum CarType  
{  
    Toyota = 1,  
    Honda = 2,  
    Ford = 3,  
}
```

- <https://docs.microsoft.com/en-us/dotnet/csharp/language-reference/keywords/switch>
- [https://www.learncs.org/en/Variables\\_and\\_Types](https://www.learncs.org/en/Variables_and_Types)

# ДНК кодирование

```
public enum DNACode
{
    A,
    G,
    T,
    C
}
```

# ДНК кодирование

```
public enum DNACode
{
    A,
    G,
    T,
    C
}
```

```
namespace DNACode
{
    2 references
    public enum DNACode { A, G, T, C }

    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            DNACode code = DNACode.A;
            Console.WriteLine("Hello DNA World!");
        }
    }
}
```

# задача: кодирование ДНК

## задача 3

вывести на экране значение кода аминокислоты

# СИНТАКСИС

- переменные
- **приведение типов**
- if/else/ ?:
- switch
- while, do{}while()
- for, foreach
- IO: WriteLine, ReadLine
- methods

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>

# приведение типов

- Неявные преобразования

```
int num = 2147483647;  
long bigNum = num;
```

- Явные преобразования

```
double x = 1234.7;  
int a;  
a = (int)x;
```

- Пользовательские преобразования
- Преобразования с использованием вспомогательных классов

<https://docs.microsoft.com/ru-ru/dotnet/csharp/programming-guide/types/casting-and-type-conversions>



# задача: ДНК кодирование

```
namespace DNACode
{
    2 references
    public enum DNACode { A, G, T, C }

    0 references
    class Program
    {
        0 references
        static void Main(string[] args)
        {
            DNACode code = DNACode.C;
            Console.WriteLine("DNA code is {0}", (int)code);
        }
    }
}
```

Microsoft Visual Studio Debug Console

DNA code is 3

C:\Program Files\dotnet\dotnet.exe (process 21688) exited with code 0.  
To automatically close the console when debugging stops, enable Tools->Options->Debug Console->Close console when debugging stops.  
Press any key to close this window . . .

# задача: кодирование ДНК

## задача 3

вывести на экране значение кода аминокислоты  
в зависимости от самой аминокислоты

# СИНТАКСИС

- переменные
- приведение типов
- **if/else/ ?:**
- **switch**
- while, do{}while()
- for, foreach
- IO: WriteLine, ReadLine
- methods

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>

# if, else, ?:

```
int a = 4;
int b = 5;
bool result;
result = a < b; // true
result = a > b; // false
result = a <= 4; // a smaller or equal to 4 - true
result = b >= 6; // b bigger or equal to 6 - false
result = a == b; // a equal to b - false
result = a != b; // a is not equal to b - true
result = a > b || a < b; // Logical or - true
result = 3 < a && a < 6; // Logical and - true
result = !result; // Logical not - false
```

```
if (a == b) {
    // We already know this part
} else {
    // a and b are not equal... :/
}
```

```
int a = 1;
int b = 2;
int c = a == b ? 100 : 200;
```

The conditional operator `?:`, commonly known as the ternary conditional operator

# множественный if

```
if (Condition1)
{ // cond1
}
else if (Condition2)
{ // Condition1 is false and Condition2 is true.
}
else if (Condition3)
{ // Condition1,2 are false and Condition3 is true.
}
else
{ // else
}
```

# switch

```
int x = 10;

switch (x)
{
    case 5:
        Console.WriteLine("Value of x is 5");
        break;
    case 10:
        Console.WriteLine("Value of x is 10");
        break;
    case 15:
        Console.WriteLine("Value of x is 15");
        break;
    default:
        Console.WriteLine("Unknown value");
        break;
}
```

# задача: кодирование ДНК

## задача 4

вывести на экран последовательность кодов аминокислот

# СИНТАКСИС

- переменные
- приведение типов
- if/else/ ?:
- switch
- **while, do{}while()**
- **for, foreach**
- IO: WriteLine, ReadLine
- methods

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>



# while

```
while([conditions to be checked])  
{  
  [Code to execute]  
}
```

```
int n = 0;  
  
while( n == 0)  
{  
    Console.WriteLine("N is 0");  
    n++;  
}
```

links: - [https://www.learncs.org/en/While\\_loops](https://www.learncs.org/en/While_loops)  
- <http://www.tutorialspoint.com/csharp/index.htm>

# do{} while()

```
do
{
    //execute code block

} while(boolean expression);
```

```
int i = 0;

do
{
    Console.WriteLine("Value of i: {0}", i);

    i++;

} while (i < 10);
```

links: - <https://www.tutorialsteacher.com/csharp/csharp-do-while-loop>  
- <http://www.tutorialspoint.com/csharp/index.htm>

# for

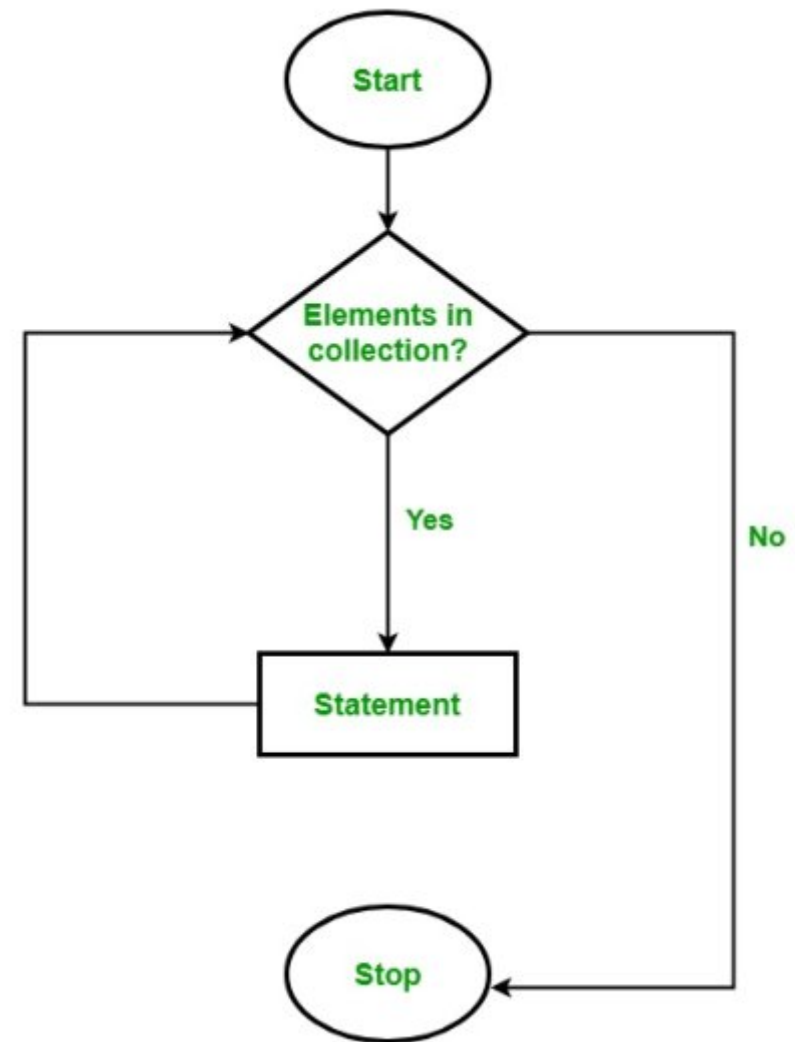
```
for( [variable to count iterations] ; [conditions checked for] ; [code to execute every loop])  
{  
  
}
```

```
for(int i = 0; i < 16; i++)  
{  
  
    if(i == 12)  
    {  
        break;  
    }  
  
}
```

links: - [https://www.learncs.org/en/For\\_loops](https://www.learncs.org/en/For_loops)  
- <http://www.tutorialspoint.com/csharp/index.htm>

# foreach

```
var names = new string[] { "John", "Tom", "Peter" };  
foreach (string name in names)  
{  
    Console.WriteLine(name);  
}
```



links: - <http://www.csharp-examples.net/foreach/>  
- <http://www.tutorialspoint.com/csharp/index.htm>

# foreach

```
0 references
static void Main(string[] args)
{
    string s = "hey! it is a string";

    int l = 0;
    foreach(char c in s)
    {
        string pad = "";
        for (int i = 0; i < l; i++)
            pad += " ";
        l += 1;

        Console.WriteLine(pad + c);
    }
}
```

Microsoft Visual Studio Deb

h  
e  
y  
!  
  
i  
t  
  
i  
s  
  
a  
  
s  
t  
r  
i  
n  
g

links: - [https://www.learncs.org/en/For\\_loops](https://www.learncs.org/en/For_loops)  
- <http://www.tutorialspoint.com/csharp/index.htm>

# СИНТАКСИС

- переменные
- приведение типов
- if/else/ ?:
- switch
- while, do{}while()
- for, foreach
- **IO: Write, WriteLine, ReadLine**
- methods

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>

# WriteLine/ReadLine

```
string line;  
while ((line = Console.ReadLine()) != null)  
{  
    if (line == "exit")  
        break;  
    Console.WriteLine(line);  
}
```

links: - <https://docs.microsoft.com/en-us/dotnet/api/system.console.readline?view=netframework-4.8>  
- <http://www.tutorialspoint.com/csharp/index.htm>

# СИНТАКСИС

- переменные
- приведение типов
- if/else/ ?:
- switch
- while, do{}while()
- for, foreach
- IO: Write, WriteLine, ReadLine
- **methods**

links: - <https://www.learncs.org/>

- <http://www.tutorialspoint.com/csharp/index.htm>



# methods

```
[Modifiers (E.G public or static)] [Type of output] [Name] ( [parameter 1],[parameter 2] ... )  
{  
  
}
```

```
public static int Multiply(int a, int b)  
{  
    return a * b;  
}
```

links: - <https://www.learncs.org/en/Methods>  
- <http://www.tutorialspoint.com/csharp/index.htm>

# arrays

```
[Modifiers (E.G public or static)] [Type of output] [Name] ( [parameter 1],[parameter 2] ... )  
{  
  
}
```

```
public static int Multiply(int a, int b)  
{  
    return a * b;  
}
```

links: - <https://www.learncs.org/en/Arrays>

- <http://www.tutorialspoint.com/csharp/index.htm>

# задача: кодирование ДНК

строка  $S$  - последовательность ДНК

$S = \langle l_1, l_2, l_3, \dots, l_n \rangle$

$l_i \in \{0, 1, 2, 3\}, l_i \in \{'A', 'G', 'T', 'C'\}$

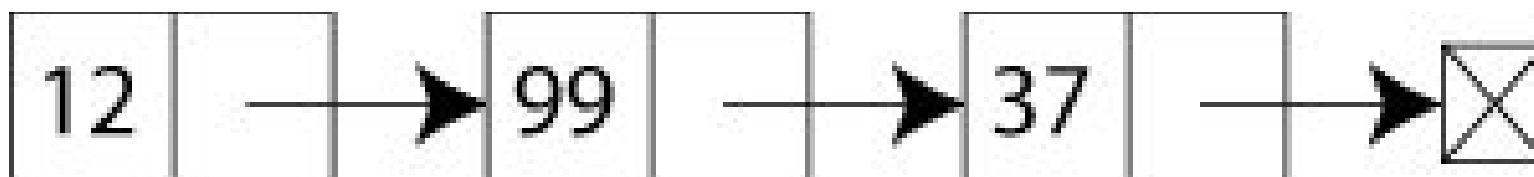
задача: по входной строке вида "021123" вывести

1. последовательность "ATGGTC"

2. цепочку полных названий

"Adenin-Thymine-Guanine-Guanine-Thymine-Cytosine"

# Списки - Lists



```
using System;  
using System.Collections.Generic;
```

```
List<int> seq = new List<int>();  
seq.Add(12);  
seq.Add(99);  
seq.Add(37);  
  
int e = seq[2];
```

# Списки - Lists

```
List<string> fruits = new List<string>();  
// add fruits  
fruits.Add("apple");  
fruits.Add("banana");  
fruits.Add("orange");  
  
// now remove the banana  
fruits.RemoveAt(1);  
Console.WriteLine(fruits.Count);
```

```
List<string> food = new List<string>();  
food.Add("apple");  
food.Add("banana");
```

```
List<string> vegetables = new List<string>();  
vegetables.Add("tomato");  
vegetables.Add("carrot");
```

```
food.AddRange(vegetables);  
Console.WriteLine(food.Count);
```

# Словари - Dictionaries

```
Dictionary<string, long> phonebook = new Dictionary<string, long>();  
phonebook.Add("Alex", 4154346543);  
phonebook["Jessica"] = 4159484588;
```

```
Dictionary<string, long> phonebook = new Dictionary<string, long>();  
phonebook.Add("Alex", 415434543);  
phonebook["Jessica"] = 415984588;
```

```
if (phonebook.ContainsKey("Alex"))  
{  
    Console.WriteLine("Alex's number is " + phonebook["Alex"]);  
}
```



# Словари - Dictionaries

```
Dictionary<string, long> phonebook = new Dictionary<string, long>();  
phonebook.Add("Alex", 415434543);  
phonebook["Jessica"] = 415984588;  
  
phonebook.Remove("Jessica");  
Console.WriteLine(phonebook.Count);
```

# String.Split

```
string s = "just a string";  
char last = s[s.Length - 1];  
  
string[] res = s.Split(" ");  
foreach (var sub in res)  
    Console.WriteLine(sub);
```



# String.Split

```
string s = "just a string";  
char last = s[s.Length - 1];  
  
string[] res = s.Split(" ");  
foreach (var sub in res)  
    Console.WriteLine(sub);
```