

7-ші дәріс. Енгізу-шығару және басқару операторлары элементтеріне мысалдар

- Консольдік енгізу-шығару (Console класы) мүмкіндіктері;
- Басқару операторлары элементтері (тармақ, цикл, басқаруды беру).

Консольге мәлімет шығару

```
using System;
namespace A
{
    class Class1
    {
        static void Main()
        {
            int i = 3;
            double y = 4.12;
            decimal d = 600m;
            string s = "Берік";

            Console.Write( i );
            Console.Write( " y = {0:F2} \nd = {1:D3}", y, d );
            Console.WriteLine( " s = " + s );
        }
    }
}
```

Консольден мәлімет енгізу

```
using System;
namespace A
{
    class Class1
    {
        static void Main()
        {
            string s = Console.ReadLine();           // ввод строки

            char c = (char)Console.Read();           // ввод символа
            Console.ReadLine();

            stringbuf;                               // буфер для ввода чисел
            buf = Console.ReadLine();

            int i = Convert.ToInt32( buf );          // преобразование в целое
            buf = Console.ReadLine();
            double x = Convert.ToDouble( buf );      // преобразование в вещ.
            buf = Console.ReadLine();
            double y = double.Parse( buf );          // преобразование в вещ.
        }
    }
}
```

■ Математикалық функциялар: Math класы

Аты	Сипаты	Нәтижесі	Түсініктеме
Abs	Модуль	асыра жүктелген	$ x \rightarrow \text{Abs}(x)$

Acos	Арккосинус	double	Acos(double x)
Asin	Арксинус	double	Asin(double x)
Atan	Арктангенс	double	Atan(double x)
Atan2	Арктангенс	double	Atan2(double x, double y)—y/x мәніне тең бұрыш тангенсі
BigMul	Көбейту	long	BigMul(int x, int y)
Ceiling	Үлкен бүтінге дейін дөңгелектеу	double	Ceiling(double x)
Cos	Косинус	double	Cos(double x)
Cosh	Гиперболалық косинус	double	Cosh(double x)
DivRem	Бөлу және қалдық	асыра жүктелген	DivRem(x, y, rem)
E	Натурал логарифм негізі (e саны)	double	2,71828182845905
Exp	Экспонента	double	$e^x \rightarrow \text{Exp}(x)$
Floor	Кіші бүтінге дейін дөңгелектеу	double	Floor(double x)
IEEE Remainder	Бөлгендегі қалдық	double	IEEERemainder(double x, double y)
Log	Натурал логарифм	double	$\log x \rightarrow \text{Log}(x)$
Log10	Ондық логарифм	double	$\log_{10} x \rightarrow \text{Log10}(x)$
Max	Екі сан максимумы	асыра жүктелу	Max(x, y)
Min	Екі сан минимумы	перегружен	Min(x, y)
PI	π саны мәні	double	3,14159265358979
Pow	Дәрежелеу	double	$x^y \rightarrow \text{Pow}(x, y)$
Round	Дөңгелектеу	асыра жүктелу	Round(3.1) нәтижесі 3 Round (3.8) нәтижесі 4
Sign	Сан таңбасын табу	int	Аргументтері асыра жүктелген
Sin	Синус	double	Sin(double x)

Sinh	Гиперболалық синус	double	Sinh(double x)
Sqrt	Квадрат түбір	double	$\sqrt{x} \rightarrow \text{Sqrt}(x)$
Tan	Тангенс	double	Tan(double x)
Tanh	Гиперболалық тангенс	double	Tanh(double x)

Мысал: температураны Фаренгейттен (F) Цельсияға (C) көшіру программасы

using System;

namespace CA1

{ class Class1

{ static void Main()

{

Console.WriteLine("Введите температуру по Фаренгейту");

double fahr = Convert.ToDouble(Console.ReadLine());

double cels = 5.0 / 9 * (fahr - 32);

Console.WriteLine("По Фаренгейту: {0} в градусах Цельсия: {1}",
fahr, cels);

}

}

}

$$C = \frac{5}{9} (F - 32)$$

Басқару операторларының элементтері

Блок (құрама оператор)

Блок — жүйелі жақшаларға алынған операторлар тізбегі:

begin end

{ }

Блок компилятор үшін бір оператор болып саналады, синтаксис бойынша бір оператор керек болғанмен, алгоритм бойынша — бірнеше оператор орындалады.

Блокта бір-ақ оператор болуы мүмкін, кейде ол тіпті бос болады.

«Өрнек» операторы

Нүктелі үтірмен аяқталған кез келген өрнек оператор болып есептеледі, ол белгілі бір амалдар орындауды керек етеді.

i++; // выполняется операция инкремента

a *= b + c; // выполняется умножение с присваиванием

fun(i, k); // выполняется вызов функции

Бос оператор

○ *Бос оператор* ; синтаксис бойынша оператор керек болғанмен, мағынасы бойынша — ол қажет етілмегенде қолданылады:

○ while (true);

Бос оператордан тұратын бұл цикл шексіз орындалуды көрсетеді

○ ;;;

Үш бос оператор

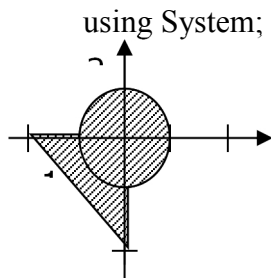
Тармақталу операторы:

- тармақталу (if)
- ауыстыру (switch)

```

if шартты операторы
if ( a < 0 ) b = 1;
if ( a < b && ( a > d || a == 0 ) ) ++b;
else { b *= a; a = 0; }
if ( a < b ) if ( a < c ) m = a;
                else m = c;
else if ( b < c ) m = b;
                else m = c;
    
```

Мысал



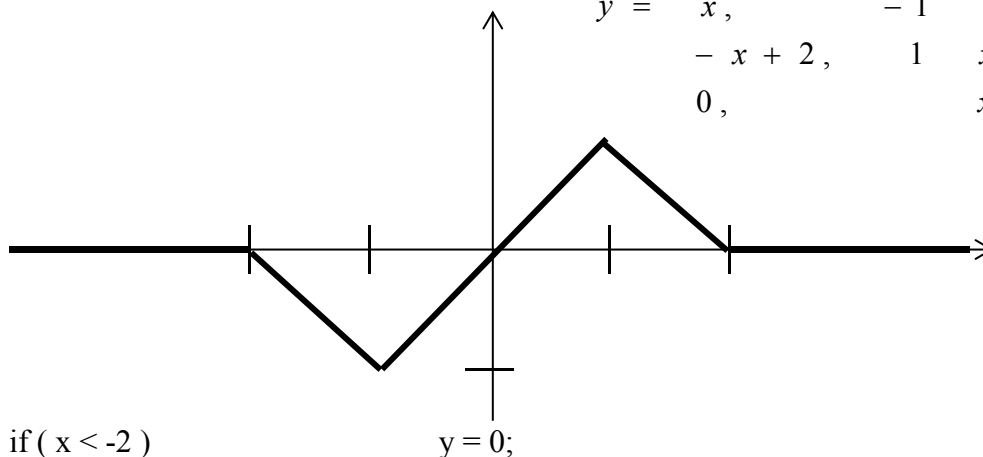
```

using System;

namespace ConsoleApplication1
{
    class Class1
    {
        static void Main()
        {
            Console.WriteLine( "Введите координату x" );
            double x = Convert.ToDouble( Console.ReadLine() );
            Console.WriteLine( "Введите координату y" );
            double y = double.Parse( Console.ReadLine() );
            if ( x * x + y * y <= 1 ||
                x <= 0 && y <= 0 && y >= - x - 2 )
                Console.WriteLine( "Точка попадает в область" );
            else
                Console.WriteLine( "Точка не попадает в область" );
        }
    }
}
    
```

2 мысал

$$y = \begin{cases} 0, & x < -2 \\ -x - 2, & -2 \leq x < -1 \\ x, & -1 \leq x < 1 \\ -x + 2, & 1 \leq x < 2 \\ 0, & x \geq 2 \end{cases}$$



```
else if ( x < -1 )    y = -x - 2;
  else if ( x <  1 )    y = x;
  else if ( x <  2 )    y = -x + 2;
  else                  y = 0;
```

```
if      ( x <= -2 )    y = 0;
if ( x >= -1 && x <  1 ) y = x;
if ( x >=  1 && x <  2 ) y = -x + 2;
if ( x >=  2 )          y = 0;
```

```
y = 0;
if ( x > -2 ) y = -x - 2;
if ( x > -1 ) y = x;
if ( x >  1 ) y = -x + 2;
if ( x >  2 ) y = 0;
```