



Увлекательное программирование на C#

Работа с черепашкой

Turtle.Move(100) – движение вперед

Turtle.Turn(90) – поворот

Turtle.PenUp() / Turtle.PenDown() – поднять / опустить перо

Turtle.Speed = 10 – задать скорость

Чтобы команды черепашки и рисования стали доступны, необходимо добавить ссылку (References) на Smallbasic.dll и написать в программе using Microsoft.SmallBasic.Library

Рисование по координатам

GraphicsWindow.Show() - показать окно

GraphicsWindow.DrawRectangle(x,y,w,h) – рисовать прямоугольник

GraphicsWindows.DrawEllipse/FillEllipse – рисовать эллипс или окружность (без заливки/с заливкой)

GraphicsWindow.DrawLine – рисовать линию

GraphicsWindows.PenColor="Red" – установить цвет

Работа с консолью

Console.Write("...") / WriteLine("..") – печать сообщения (без или с переводом строки)

Console.Write("X={0}",x) – печать по шаблону

var x = Console.ReadLine() – считывание строки с консоли

var x = Console.ReadKey() – считывание нажатия одной клавиши

Чтобы команды работы с консолью стали доступны, в начале программы должно быть написано using System

Правила синтаксиса

;- разделение операторов

{ ... } – несколько операторов группируются вместе

Описание функции

```
static void square(int x)
{
    <тело функции>
}
```

Параметры

Тип
возвращаемого
значения

Условный оператор

if (условие) then оператор;

if (условие) then оператор1

else оператор2;

Условие:

- Проверка, $x > 0$, $z == 0$ или $u != v$
- $y1 \ \&\& \ y2$ – оба условия выполняются
- $y1 \ || \ y2$ – одно из условий
- $!y$ – условие не выполняется

Циклы

while (условие) do оператор;

for (инициализатор; условие; инкремент)
оператор1

Повторение от 0 до 9:

for (int i=0; i<10; i++) { ... }

Типы данных

int – целый

float / double – вещественный

string – строка

Математика

Math.sin, cos, ...

Math.sqrt – квадратный корень

Массивы

int[] arr – объявление

arr = new int[5] – выделение
памяти

arr[i] – обращение к элементу

Классы и объекты

class House

```
{
    public static int cnt=0; // стат.поле
    public int x, y, size; // поля
    public House(int cx, int cy, int csz)
    { ... } // конструктор
    public string Color
    { // свойство
        set { ... }
        get { return color; } }
    public void Draw() // метод
    { .... }
}
```