C# basics

Introduction to Computer Programming

Maciej Piernik

Review of Lecture 1

0 Variables

0 Basic types

0 Basic interaction with user

Outline

0 Operators

0 Conditional statements

0 Loops

0 Excercises

Operators

○ Assignment

```
int x = 23;
x = x + 1;
```

Assignment returns a value!
```
int y = x = x + 5;
```

○ Rzutowanie

```
float f = 23;
int i = (int)f;
```

Operators

○ Arithmetic

```
z = x + y;
z = x - y;
z = x * y;
z = x / y;
z = x % y;

x++;
x--;
++x;
--x;
```

○ String concatenation "str1" + "str2"

```
string a = "Bond, ";
Console.WriteLine(a + "James Bond");
```

Operator

- Comparison

```
x > y
x < y
x >= y
x <= y
x == y
x != y
```

- Logical and conditional logical

```
p & q
p | q
p ^ q
p && q
p || q
!p
```

Operators

O Assignment cont.

```
x += y; //x = x + y;
x -= y; //x = x - y;
x *= y; //x = x * y;
x /= y; //x = x / y;
x %= y; //x = x % y;

p &= q; //p = p & q;
p |= q; //p = p | q;
p ^= q; //p = p ^ q;
```

Conditional statements - if

```
if (logical condition)
{
    instructions...
}
```

Conditional statements - if

```csharp
int points = int.Parse(Console.ReadLine());

if (points > 5)
{
    Console.WriteLine("Congratulations! You passed!");
}

Console.WriteLine("Your score is: " + points);
```

{ } = block of code

Conditional statements - if

```csharp
int points = int.Parse(Console.ReadLine());

if (points > 5)
{
    Console.WriteLine("Congratulations! You passed!");
}
else
{
    Console.WriteLine("Unfortunately, you didn't pass.");
}

Console.WriteLine("Your score is: " + points);
```

Conditional statements - if

```csharp
int points = int.Parse(Console.ReadLine());

if (points == 10)
{
    Console.WriteLine("Great score!!!");
}
else if (points > 5)
{
    Console.WriteLine("Congratulations! You passed!");
}
else
{
    Console.WriteLine("Unfortunately, you didn't pass.");
}

Console.WriteLine("Your score is: " + points);
```

Conditional statements
switch

```
switch (logical condition)
{
    case value 1:
        instructions...
        break;
    case value 2:
        instructions...
        break;
    ...
    default:
        instructions...
        break;
}
```

Conditional statements
switch

```csharp
int points = int.Parse(Console.ReadLine());

Console.Write("Your grade: ");
switch (points)
{
    case 10:
        Console.WriteLine("Excellent");
        break;
    case 9:
        Console.WriteLine("Very good");
        break;
    case 8:
        Console.WriteLine("Good");
        break;
    case 7:
        Console.WriteLine("Satisfactory");
        break;
    case 6:
        Console.WriteLine("Sufficient");
        break;
    default:
        Console.WriteLine("Fail");
        break;
}
```

Conditional statements
switch

```csharp
Console.Write("Select an option: n: new game, s: settings, q: quit > ");
string option = Console.ReadLine();

switch (option)
{
    case "n":
        Console.WriteLine("Let's start the game!");
        break;
    case "s":
        Console.WriteLine("Settings...");
        break;
    case "q":
        Console.WriteLine("Bye bye!");
        break;
    default:
        Console.WriteLine("There's no such option!");
        break;
}
```

Loops – `while`

```
while (logical condition)
{
    instructions...
}
```

Loops – while

```csharp
int l = 5;

while (l > 0)
{
    Console.WriteLine(l);
    l--;
}
```

Loops – `while`

```csharp
Console.WriteLine("Welcome to my division program!");
Console.Write("Numerator: ");
int numerator = int.Parse(Console.ReadLine());

int denumerator = 0;

while (denumerator == 0)
{
    Console.Write("Denumerator: ");
    denumerator = int.Parse(Console.ReadLine());
}

Console.WriteLine("Result: " + ((float)numerator / denumerator));
```

Loops – `while`

```csharp
Console.WriteLine("Welcome to my division program!");
Console.Write("Numerator: ");
int numerator = int.Parse(Console.ReadLine());

int denumerator = 0;

do
{
    Console.Write("Denumerator: ");
    denumerator = int.Parse(Console.ReadLine());
} while (denumerator == 0);

Console.WriteLine("Result: " + ((float)numerator / denumerator));
```

TryParse

```
int variable = 0;

bool success = int.TryParse("text to parse", out variable);
```

Loops – `while`

```
Console.WriteLine("Welcome to my division program!");
Console.Write("Numerator: ");
int numerator = int.Parse(Console.ReadLine());

int denumerator = 0;

do
{
    Console.Write("Denumerator: ");
} while (!int.TryParse(Console.ReadLine(), out denumerator) || denumerator == 0);

Console.WriteLine("Result: " + ((float)numerator / denumerator));
```

Loops – for

```
for (initialization; logical condition; update)
{
    instructions...
}
```

Loops – for

```csharp
for (int i = 0; i < 10; i++)
{
    Console.WriteLine(i);
}
```

Loops – for

```csharp
Console.Write("How many numbers should I write: ");

int numbers = int.Parse(Console.ReadLine());

for (int i = 0; i < numbers; i++)
{
    Console.WriteLine(i);
}
```

Summary

O Operators

O Conditional statements

O Loops