Khoa Công nghệ Thông tin

NGÔN NGỮ LẬP TRÌNH C#

Mục tiêu bài học

Khoa Công nghệ Thông tin

Nắm được các khái niệm trong ngôn ngữ lập trình C#

- Biến
- Kiểu dữ liệu
- Cấu trúc điều khiển: lặp, điều kiện,..
- …

Nội dung

Khoa Công nghệ Thông tin

- Giới thiệu C#
- Kiểu dữ liệu
- Biến, hằng
- Ép kiểu
- Nhập xuất trong C#
- Cấu trúc điều kiện
- Cấu trúc lặp
- Chương trình con
- ….

Giới thiệu về ngôn ngữ lập trình C#

- Visual C# được Mircosoft công bố 2002 là ngôn ngữ mạnh, đơn giản
- Đặc điểm:
  - Có nguồn gốc C, C++
  - Ngôn ngữ hướng đối tượng: đa hình, kế thừa, đóng gói
  - Có thể xây dựng nhiều loại ứng dụng: web, dịch vụ web, đồ họa,..

Từ khóa C#

Khoa Công nghệ Thông tin

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

Từ khóa C#

Khoa Công nghệ Thông tin

- namespace: khai báo một không gian tên.
  - Được thiết kế bởi các class có thuộc tính và phương thức => tránh trùng lặp tên
- static: thường đặt trước khai báo hàm, có thể truy xuất mà không cần tạo đối tượng của lớp.
- using: dùng để khai báo các namespace, giúp chương trình gọn nhẹ.
- VD:

```
System.Windows.Form.Button bt = new System.Windows.Form.Button ();
=>
using System.Windows.Form;
//...
Button bt = new Button ();
```

Kiểu dữ liệu

Khoa Công nghệ Thông tin

- Trong .NET
  - Mọi kiểu đều là đối tượng
  - Gốc của mọi đối tượng là lớp System.Object
  - .NET có các kiểu: Byte, Char, Int32, String,…
  - Mỗi ngôn ngữ lập trình (C#, VB.NET,J#,..) sẽ có tên kiểu ánh xạ tương ứng. Có thể sử dụng thay đổi cho nhau.

Kiểu dữ liệu

Khoa Công nghệ Thông tin

- Các kiểu dữ liệu cơ bản

C# type	Size (in bytes)	.NET type	Description
byte	1	Byte	Unsigned (values between 0 and 255).
char	2	Char	Unicode characters (a modern way of storing most characters, including international language characters).
bool	1	Boolean	True or false.
sbyte	1	SByte	Signed (values between −128 and 127).
short	2	Int16	Signed (short) (values between −32,768 and 32,767).
ushort	2	UInt16	Unsigned (short) (values between 0 and 65,535).
int	4	Int32	Signed integer values between −2,147,483,648 and 2,147,483,647.
uint	4	UInt32	Unsigned integer values between 0 and 4,294,967,295.
float	4	Single	Floating-point number. Holds the values from approximately +/−1.5×10^{-45} to approximately +/−3.4×10^{38} with seven significant figures.
double	8	Double	Double-precision floating-point. Holds the values from approximately +/−5.0×10^{-324} to approximately +/−1.8×10^{308} with 15 to 16 significant figures.
decimal	12	Decimal	Fixed-precision up to 28 digits and the position of the decimal point. This type is typically used in financial calculations. Requires the suffix "m" or "M" when you declare a constant.
long	8	Int64	Signed integers ranging from −9,223,372,036,854,775,808 to 9,223,372,036,854,775,807.
ulong	8	UInt64	Unsigned integers ranging from 0 to approximately 1.85×10^{19}.

Kiểu dữ liệu

Khoa Công nghệ Thông tin

- Kiểu dữ liệu cấu trúc: từ khóa struct

```
struct time
{
    public int hours, minutes, seconds;
}
```

- Kiểu mảng:

```
String[] st = new String[3];
st[0] = "C";
st[1] = "B";
st[2] = "A";
```

Các toán tử

Khoa Công nghệ Thông tin

Phép toán	Kí hiệu
So sánh	==, !=, >,<, >=, <=
Số học	+, - ,*, /, %
Gán	=, +=, -=, *=, /=, %=
Logic	&&, ||, !, true, false
Ép kiểu	( )
Tăng giảm	++ --
Điều kiện	? : (vd: 3>0?1:0)
Truy nhập thành viên	.
Chỉ số	[ ]
Tạo đối tượng	new

Ép kiểu

Khoa Công nghệ Thông tin

- Cách 1: Dùng phương thức Parse của đối tượng kiểu
  - VD:

    String s="12";

    int b=int.Parse(s);
- Cách 2: Dùng các phương thức của lớp Convert:
  - VD:

    String s="12";

    int b=Convert.ToInt32(s);
- Cách 3 Ép kiểu cho đối tượng dùng toán tử ()
  - VD:

    float x=123.4;

    int a = int(x);// a=123

- Chú thích:
  - // 1 dòng
  - /* */ nhiều dòng
- Khai báo sử dụng namespace

  using <tên namespace>;
  - VD:

    using System.Data;

    using System.Windows.Form;

Biến, hằng số

Khoa Công nghệ Thông tin

- Khai báo biến:

  <kiểu truy xuất> <kiểu dữ liệu> <tên biến> [=giá trị];

- Trong đó:

  - kiểu truy xuất:
    - public – bất kì đâu
    - protected – bên trong một lớp nơi biến được định nghĩa hoặc từ các lớp con của lớp đó.
    - private – chỉ truy xuất ở bên trong lớp nơi nó được định nghĩa

- Khai báo hằng:

  const <kiểu dữ liệu> <tên hằng> = giá trị;

- VD:

  public int a;

  const float p =3.14;

Nhập xuất trong C#

Khoa Công nghệ Thông tin

- Nhập xuất được thực hiện qua lớp Console của namepace System.
  - Console.Write() - Console.WriteLine(): hiện ra màn hình
  - Console.Read() - Console.ReadLine(): đọc dữ liệu từ bàn phím.

Cấu trúc điều khiển if

Khoa Công nghệ Thông tin

- Cú pháp:

```
if (biểu thức điều kiện)
{
      // câu lệnh thực thi nếu biểu thức điều kiện đúng
}
[else
{
      // câu lệnh thực thi nếu biểu thức điều kiện sai

}]
```

- Ví dụ:

```
if (20 % 4 > 0)
          {
              Console.WriteLine("Số 20 không chia hết cho 4");
          }
          else
          {
              Console.WriteLine("Số 20 chia hết cho số 4");

          }
```

Cấu trúc điều khiển switch - case

Khoa Công nghệ Thông tin

- Cú pháp:

```
switch (variable)
{
        case value:
              Câu lệnh thực thi
              break;
        case value:
              Câu lệnh thực thi
              break;
        case value:
              Câu lệnh thực thi
              break;
        default:
              Câu lệnh thực thi
        break;

}
```

- Ví dụ:

```
int x = 20 % 4;
        switch (x)
        {
            case 1:
                Console.WriteLine("20 chia cho 4 được số dư là 1");
                break;
            case 0:
                Console.WriteLine("20 chia hết cho 4");
                break;
            default:
                Console.WriteLine("Không thuộc tất cả các trường hợp trên");
                break;

        }
```

Cấu trúc lặp - while

Khoa Công nghệ Thông tin

- Cú pháp:

```
while     (condition)
{
          // câu lệnh
}
```

- Ví dụ:

```
using System;
class WhileTest
{
   public static void Main()
   {
      int n = 1;

      while (n < 6)
      {
         Console.WriteLine("Current value of n is {0}", n);
         n++;
      }
   }
}
```

Cấu trúc lặp do - while

Khoa Công nghệ Thông tin

- Cú pháp:

```
do
{
      // câu lệnh
}
while(biểu thức điều kiện)
```

- Ví dụ:

```
using System;
public class TestDoWhile
{
   public static void Main ()
   {
      int x;
      int y = 0;

      do
      {
         x = y++;
         Console.WriteLine(x);
      }

      while(y < 5);
   }
}
```

Cấu trúc lặp for

Khoa Công nghệ Thông tin

- Cú pháp:

```
for (bt khởi tạo; bt điều kiện; bt tăng/giảm)
{
    // câu lệnh
}
```

- Ví dụ:

```
using System;
public class ForLoopTest
{
    public static void Main()
    {
        for (int i = 1; i <= 5; i++)
            Console.WriteLine(i);
    }
}
```

Cấu trúc lặp foreach

- Duyệt các phần tử của mảng, tập hợp
- Không có biến đếm
- Cú pháp:

```
foreach (object in <mảng/tập hợp>)
{
        // câu lệnh
}
```

- Ví dụ:

```
int[ ] a={1,2,3,4};
int sum = 0;
foreach(int n in a)
{
    sum=sum+n;
}
```

Lệnh break, continue, return

- **break**: thoát khỏi vòng lặp và thực hiện lệnh kế tiếp sau vòng lặp
- **continue**: ngừng thực hiện các việc của vòng lặp hiện tại và quay về đầu vòng lặp để thực hiện bước lặp tiếp theo.
- **return**: thoát khỏi hàm, trả quyền điều khiển về nơi gọi hàm

Xử lý ngoại lệ - try … catch …finally

Khoa Công nghệ Thông tin

- Cú pháp:

```
try
{
    //câu lệnh
}
catch (Exception)
{
    //lệnh xử lý lỗi
}
finally
{
    //lệnh luôn được thực thi
}
```

- Ví dụ:

Hai lệnh này sẽ sinh ra lỗi nếu giá trị nhập vào trong 2 textbox không phải là số

```
try
{
    int so1 = int.Parse(txtSo1.Text);
    int so2 = int.Parse(txtSo2.Text);

    int ketqua = so1 / so2;
}
catch (FormatException)
{
    MessageBox.Show("Bản phải nhập hai số");
}
catch (DivideByZeroException)
{
    MessageBox.Show("Lỗi chia cho 0");
}
```

Lệnh này sẽ sinh ra lỗi nếu so2 = 0

Một số lớp ngoại lệ

Tên ngoại lệ	Mô tả
MethodAccessException	Lỗi truy cập, do truy cập đến thành viên hay phương thức không được truy cập
ArgumentException	Lỗi tham số đối mục
ArgumentNullException	Đối mục Null, phương thức được truyền đối mục null không được chấp nhận
ArithmeticException	Lỗi liên quan đến các phép toán
ArrayTypeMismatchException	Kiểu mảng không hợp, khi cố lưu trữ kiểu không thích hợp vào mảng
DivideByZeroException	Lỗi chia zero
FormatException	Định dạng không chính xác một đối mục nào đó
IndexOutOfRangeException	Chỉ số truy cập mảng không hợp lệ, dùng nhỏ hơn chỉ số nhỏ nhất hay lớn hơn chỉ số lớn nhất của mảng

Lớp String

Khoa Công nghệ Thông tin

- Khai báo chuỗi

```
string newString = “Chuoi muon nhan”;
string newString = “Day la \n chuoi hang”;
string newString = “Day la \t chuoi hang”;
string newString = “C:\\Data\\vidu.cs”;
string newString = @“C:\Data\vidu.cs”;
string newString = @“Dong 1
                    Dong 2”;
```

- Sử dụng ToString()

```
int i = 9;
string s = i.ToString();
```

Lớp String

- Định dạng chuỗi in ra sử dụng phương thức Format

```
string myStr = string.Format("Xe {0} mau {1}", "Toyota", "do");
string myStr = string.Format("{0:C}", 123.45);
string myStr = string.Format("{0:N}", 123456);
string myStr = string.Format("{0:P}", .3);
string myStr = string.Format("Phone number: {0:(###)-###-####}",1234567890);
```

Bài tập

Khoa Công nghệ Thông tin

- Viết chương trình nhập số nguyên dương n và hiện giá trị từ 0 đến n ra màn hình
- Giải phương trình bậc 2: ax2 + bx+c với a,b,c được nhập từ bàn phím.