

相关主题

RECOMMEND ARTICLE

- ▶ 标准C++的类型转换符
- ▶ C++数组应用之特殊矩阵的压缩存储
- ▶ 二叉树的创建、前序遍历、中序遍历、后序遍历
- ▶ 顺序队列基本操作
- ▶ 二叉树创建及遍历算法
- ▶ 关于C#3.0 新特性之对象构造者
- ▶ C#3.0 新特性之隐式类型化数组
- ▶ 利用C#实现分布式数据库查询

MORE

推荐文章

RECOMMEND ARTICLE

- ▶ 游戏音乐制作案例之《QQ三国》
- ▶ 游戏音乐制作案例之《武林外传》
- ▶ 策划与程序和美工沟通
- ▶ 龙神传说 原画
- ▶ 《LAST online》原画
- ▶ 《AION》新原画
- ▶ 数据广播方案的优化
- ▶ 网络游戏的位置同步

MORE

热门文章

HOT ARTICLE

- ▶ [电子书下载]游戏设计 — 原理与实践
- ▶ [电子书下载]网络游戏开发
- ▶ 游戏设计全过程
- ▶ [电子书下载]游戏设计技术
- ▶ [电子书下载]游戏设计理论
- ▶ CS游戏人物模型制作教程
- ▶ CG人物插画基本流程
- ▶ [转贴]MAX高级人头教程

MORE

您的位置: VC专区



文章标题	正确理解C#中的ref关键字		
来源:	[ogdev]	浏览:	[318]

最近有人问到 ref 关键字的正确用法,下面我们来举例说明。其实要更好的理解 ref 关键字,结合 C++ 代码更加容易一些。另外在开始我们的例子之前,需要提前说明几点:

C# 中的数据有两种类型:引用类型(reference types)和值类型(value types)。简单类型(包括int, long, double等)和结构(structs)都是值类型,而其他的类都是引用类型。简单类型在传值的时候会做复制操作,而引用类型只是传递引用,就像 C++ 中的指针一样。

注意 structs 在 C# 和 C++ 中的区别。在 C++ 中,structs 和类基本相同(except that the default inheritance and default access are public rather than private)。而在 C# 中,structs 和类有很大的区别。其中最大的区别(我个人觉得,同时也是容易忽略的一个地方)可能就是它是值类型,而不是引用类型。

下面这段代码是 MSDN 中的例子:

```
// cs_ref.cs
using System;
public class MyClass
{
    public static void TestRef(ref char i)
    {
        // The value of i will be changed in the calling method
        i = 'b';
    }

    public static void TestNoRef(char i)
    {
        // The value of i will be unchanged in the calling method
        i = 'c';
    }

    // This method passes a variable as a ref parameter; the value of the
    // variable is changed after control passes back to this method.
    // The same variable is passed as a value parameter; the value of the
    // variable is unchanged after control is passed back to this method.
    public static void Main()
    {
        char i = 'a'; // variable must be initialized
        TestRef(ref i); // the arg must be passed as ref
        Console.WriteLine(i);
        TestNoRef(i);
        Console.WriteLine(i);
    }
}
```

大家很容易看出输出结果是:

```
b
b
```

那么如果把这个例子做一些新的改动,将值类型(这里用的是 char)改成引用类型,程序运行又是什么效果呢?

```
// -----
// MyClass definition
public class MyClass
{
    public int Value;
}

// -----
// Tester methods
public static void TestRef(ref MyClass m)
{
    m.Value = 10;
}

public static void TestNoRef(MyClass m)
{
    m.Value = 20;
}

public static void TestCreateRef(ref MyClass m)
```

```

    m = new MyClass();
    m.Value = 100;
}

public static void TestCreateNoRef(MyClass m)
{
    m = new MyClass();
    m.Value = 200;
}

public static void Main()
{
    MyClass m = new MyClass();
    m.Value = 1;

    TestRef(ref m);
    Console.WriteLine(m.Value);

    TestNoRef(m);
    Console.WriteLine(m.Value);

    TestCreateRef(ref m);
    Console.WriteLine(m.Value);

    TestCreateNoRef(m);
    Console.WriteLine(m.Value);
}

```

大家能马上给出正确的答案么？如果能，那看来你对 ref 的用法了解得还是非常不错的。其实如果大家对 C++ 比较熟悉的话，把这段代码换成 C++ 的就好理解的多了。

```

// -----
// MyClass definition
#pragma once

class MyClass
{
public:
    int Value;
};

typedef MyClass* MyClassPtr;

// -----
// Tester methods
void TestRef(char* i)
{
    *i = 'b';
}

void TestNoRef(char i)
{
    i = 'c';
}

void TestRef(MyClassPtr* m)
{
    (*m)->Value = 10;
}

void TestNoRef(MyClassPtr m)
{
    m->Value = 20;
}

void TestCreateRef(MyClassPtr* m)
{
    delete (*m);
    *m = new MyClass();
    (*m)->Value = 100;
}

void TestCreateNoRef(MyClassPtr m)
{
    m = new MyClass();
    m->Value = 200;
}

int main(int argc, char* argv[])
{
    char c = 'a';

    TestRef(&c);
    printf("%c\n", c); // output: b
    TestNoRef(c);
    printf("%c\n", c); // output: b

    MyClassPtr m = new MyClass;
    m->Value = 1;

    TestRef(&m);
    printf("%d\n", m->Value);
}

```

```

printf("%d\n", m->Value);

TestNoRef(m);
printf("%d\n", m->Value);

TestCreateRef(&m);
printf("%d\n", m->Value);

TestCreateNoRef(m);
printf("%d\n", m->Value);

delete m;

return 0;
}

```

这两段分别用 C# 和 C++ 实现的代码的输出结果都是一样的。后面用 MyClass 测试的输出结果是：

```

10
20
100
100

```

具体的原因相信经过大家的分析应该会很清楚的。另外如果大家有兴趣可以用 structs 再试试，也可以同时对 structs 在 C++ 和 C# 中的区别有进一步的认识。

本栏目登载此文出于传递信息之目的，如有任何的问题请及时和我们联系！

无任何评论！

<p>请您注意：</p> <ul style="list-style-type: none"> <input checked="" type="checkbox"/> 尊重网上道德，遵守《全国人大常委会关于维护互联网安全的决定》及中华人民共和国其他各项有关法律法规 <input checked="" type="checkbox"/> 尊重网上道德，遵守中华人民共和国的各项有关法律法规 <input checked="" type="checkbox"/> 承担一切因您的行为而直接或间接导致的民事或刑事法律责任 <input checked="" type="checkbox"/> 中国网游研发中心新闻留言板管理人员有权保留或删除其管辖留言中的任意内容 <input checked="" type="checkbox"/> 您在中国网游研发中心留言板发表的作品，中国网游研发中心有权在网站内转载或引用 <input checked="" type="checkbox"/> 参与本留言即表明您已经阅读并接受上述条款 	<p>发表评论：</p> <p>昵 称：<input style="width: 100%;" type="text"/> <input type="button" value="GO"/></p> <p>联系EMAIL：<input style="width: 100%;" type="text"/></p> <div style="display: flex; justify-content: space-around; align-items: center;"> </div> <div style="border: 1px solid gray; height: 40px; margin-top: 5px;"></div>
---	--