http://blog.csdn.net/jipingyuan/article/details/39297409

[置顶] <详解>C语言实现程序跳转到绝对地址0x100000处执行

标签： c语言函数指针绝对地址跳转嵌入式

嵌入式笔试题：想让程序跳转到绝对地址0x100000处执行，该如何做？

网上看到有如下答案：

\*((void(\*)(void))0x100000)();

经过在VC++6.0和Linux gcc4.4.3下测试，均不能通过编译。

VC++6.0报错：error C2100: illegal indirection

GCC报错：error: void value not ignored as it ought to be

应该是怎么写呢？

经过测试，有两种方法：

答案1. (\*(void(\*)(void))0x100000)();

答案2. ((void(\*)(void))0x100000)();

仔细观察，第一种写法只是第一个\*的位置不同，第二种写法少了一个\*，但是都能正确编译通过，且正确执行。

为什么会有这两种答案呢？查阅资料后发现，与历史原因有关……

先来看看如下例子：

例一：

[cpp] view plain copy print?

#include <stdio.h>

void func(void)

{

printf("hello.\n");

}

void main(void)

{

printf("func=%d\n", func);

printf("&func=%d\n", &func);

}

运行程序后发现



两次打印结果相同！！！

按照&运算符本来的意义，它要求其操作数是一个对象，但函数名不是对象（函数是一个对象），本来&func是非法的，但很久以前有些编译器已经允许这样做，c/c++标准的制定者出于对象的概念已经有所发展的缘故，也承认了&func的合法性。

因此，对于func和&func可以这样理解，func是函数的首地址，它的类型是void ()，&func表示一个指向函数void func(void)这个对象的地址，它的类型是void (\*)()，因此func和&func所代表的地址值是一样的，但类型不一样。func是一个函数，&func表达式的值是一个指针！

既然取不取址都可以，那么\*不\*也都可以……

所以，在调用一个函数的时候，也有两种方法，正如前面的两种答案。

例二：

#include <stdio.h>

void func(void)

{

printf("hello.\n");

}

void main()

{

void (\*func\_p)(void) = func; //定义一个函数指针，这个指针无返回值，无参数，指向fun函数

(\*func\_p)();

(func\_p)();

}

上面的两种调用方法也都是正确的，编译通过，正确执行。



其实，

func\_p();

也是正确的调用方式……

更有甚者

(\*func)();

还是正确的……只是平时不这么用罢了（注意此处是func，不是func\_p）

暂且不考虑那么多调用方式（知道就好了），现在回过头来看看

(\*(void(\*)(void))0x100000)();

和

((void(\*)(void))0x100000)();

到底是什么东东……

1.首先来认识一个新的数据类型，如：void (\*)(void)，和 int\* 类似的一个数据类型，只不过int\*是一个指向int型的指针，而void (\*)(void)是一个指向函数的指针，且这个函数无返回值，无参数。

2.然后给他外层加个括号，如：(void (\*)(void))，这样是不是很像(int\*)，我们在做强制类型转换的时候需要在类型外加个括号的是吧。

3.接着把0x100000强制转化为一个函数指针，即：(void(\*)(void))0x100000

4.最后就是调用这个函数，外层再加个括号，后面在加一对括号（参考例二的形式），

如：((void(\*)(void))0x100000)();就可以到绝对地址0x100000处去执行了……

或者(\*(void(\*)(void))0x100000)();只是加不加 \* 的问题。上面例二中可以看出，在用函数指针调用一个函数时，加不加 \* 都是可以的。

所以答案就出来了……

另外，你可能疑惑，按照例二中func\_p(); 的形式，那么 (void(\*)(void))0x100000();也应该对呀？

但是，实际测试，编译报错：error C2064: term does not evaluate to a function

为啥呢？我也不知道了……反正不管有没有 \* ，记得加个括号就好了……

那为什么最开始的\*((void(\*)(void)0x100000))();不对呢？

因为没见过\*func(); 这么用的……

如果要是外面再加一层括号就对了，如：(\*( (void(\*)(void)0x100000) )) ();

其实，把蓝色括号去掉（蓝色括号和绿色括号重复了），就又变成答案一了……

所以无论如何，最外层不能是\* ，必须是括号！

因为没见过 \*func(); 这么用的……