文中\_\_FILE\_\_与示例1可以参见[《使用ANSI C and Microsoft C++中常用的预定义宏》](http://www.cnblogs.com/morewindows/archive/2011/08/17/2143523.html)

宏中的#的功能是将其后面的宏参数进行字符串化操作（Stringizing operator），简单说就是在它引用的宏变量的左右各加上一个双引号。

如定义好#define STRING(x) #x之后，下面二条语句就等价。

       char \*pChar = "hello";

       char \*pChar = STRING(hello);

还有一个#@是加单引号（Charizing Operator）

#define makechar(x)  #@x

       char ch = makechar(b);与char ch = 'b';等价。

但有小问题要注意，宏中遇到#或##时就不会再展开宏中嵌套的宏了。什么意思了？比如使用char \*pChar =STRING(\_\_FILE\_\_);虽然\_\_FILE\_\_本身也是一个宏，但编译器不会展开它，所以pChar将指向"\_\_FILE\_\_"而不是你要想的形如"D:\XXX.cpp"的源文件名称。因此要加一个中间转换宏，先将\_\_FILE\_\_解析成"D:\XXX.cpp"字符串。

定义如下所示二个宏：

#define \_STRING(x) #x

#define STRING(x) \_STRING(x)

再调用下面语句将输出带""的源文件路径

       char\* pChar = STRING(\_\_FILE\_\_);

       printf("%s %s\n", pChar, \_\_FILE\_\_);

可以比较下STRING(\_\_FILE\_\_)与\_\_FILE\_\_的不同，前将带双引号，后一个没有双引号。

再讲下##的功能，它可以拼接符号（Token-pasting operator）。

MSDN上有个例子：

#define paster( n ) printf( "token"#n" = %d\n", token##n )

int token9 = 100;

再调用  paster(9);宏展开后token##n直接合并变成了token9。整个语句变成了

printf( "token""9"" = %d", token9 );

在C语言中字符串中的二个相连的双引号会被自动忽略，于是上句等同于

printf("token9 = %d", token9);。

即输出token9 = 100

有了上面的基础后再来看示例1

#define WIDEN2(x) L ## x

#define WIDEN(x) WIDEN2(x)

#define \_\_WFILE\_\_ WIDEN(\_\_FILE\_\_)

wchar\_t \*pwsz = \_\_WFILE\_\_;

第一个宏中的L是将ANSI字符串转化成unicode字符串。如：wchar\_t \*pStr = L"hello";

再来看wchar\_t \*pwsz = \_\_WFILE\_\_;

\_\_WFILE\_\_被首先展开成WIDEN(\_\_FILE\_\_)，再展开成WIDEN2("\_\_FILE\_\_表示的字符串")，再拼接成 L"\_\_FILE\_\_表示的字符串" 即L"D:\XXX.cpp" 从而得到unicode字符串并取字符串地址赋值给pwsz指针。

在VC中\_T()，TEXT ()也是用的这种技术。

在tchar.h头文件中可以找到：

#define \_T(x)       \_\_T(x)

#define \_\_T(x)      L ## x

在winnt.h头文件中可以找到

#define TEXT(quote) \_\_TEXT(quote)   // r\_winnt

#define \_\_TEXT(quote) L##quote      // r\_winnt

因此不难理解为什么第三条语句会出错error C2065: 'LszText' : undeclared identifier

       wprintf(TEXT("%s %s\n"), \_T("hello"), TEXT("hello"));

       char szText[] = "hello";

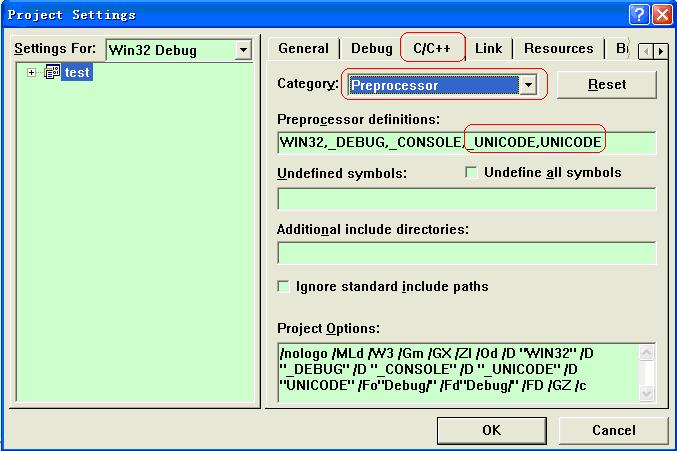
       wprintf(TEXT("%s %s\n"), \_T(szText), TEXT(szText));

而将"hello"定义成宏后就能正确运行。

#define SZTEXT "hello"

       wprintf(TEXT("%s %s\n"), \_T(SZTEXT), TEXT(SZTEXT));

注：由于VC6.0默认是ANSI编码，因此要先设置成unicode编码，在project菜单中选择Setting，再在C/C++标签对话框中的Category中选择Preprocessor。再地Preprocessor definitions编辑框中将\_MBCS去掉，加上\_UNICODE,UNICODE。



更多内容可以查考MSDN上对[Preprocessor Operators](http://msdn.microsoft.com/en-us/library/wy090hkc(v=vs.80).aspx)的讲解。

转载请标明出处，原文地址：<http://www.cnblogs.com/morewindows/archive/2011/08/18/2144112.html>