windows开发环境下用VC++6.0 对unicode 、utf-8、 gb2312 三种编码格式之间的转换方法:

**[cpp]** [view plain](http://blog.csdn.net/guzicheng/article/details/7645344) [copy](http://blog.csdn.net/guzicheng/article/details/7645344)

1. #include <iostream>
2. #include <string>
3. #include <Windows.h>
4. **using** **namespace** std;
6. **void** unicodeToUTF8(**const** wstring &src, string& result)
7. {
8. **int** n = WideCharToMultiByte( CP\_UTF8, 0, src.c\_str(), -1, 0, 0, 0, 0 );
9. result.resize(n);
10. ::WideCharToMultiByte( CP\_UTF8, 0, src.c\_str(), -1, (**char**\*)result.c\_str(), result.length(), 0, 0 );
11. }
13. **void** unicodeToGB2312(**const** wstring& wstr , string& result)
14. {
15. **int** n = WideCharToMultiByte( CP\_ACP, 0, wstr.c\_str(), -1, 0, 0, 0, 0 );
16. result.resize(n);
17. ::WideCharToMultiByte(CP\_ACP, 0, wstr.c\_str(), -1, (**char**\*)result.c\_str(), n, 0, 0 );
18. }
20. **void** utf8ToUnicode(**const** string& src, wstring& result)
21. {
22. **int** n = MultiByteToWideChar( CP\_UTF8, 0, src.c\_str(), -1, NULL, 0 );
23. result.resize(n);
24. ::MultiByteToWideChar( CP\_UTF8, 0, src.c\_str(), -1, (**LPWSTR**)result.c\_str(), result.length());
25. }
27. **void** gb2312ToUnicode(**const** string& src, wstring& result)
28. {
29. **int** n = MultiByteToWideChar( CP\_ACP, 0, src.c\_str(), -1, NULL, 0 );
30. result.resize(n);
31. ::MultiByteToWideChar( CP\_ACP, 0, src.c\_str(), -1, (**LPWSTR**)result.c\_str(), result.length());
32. }
34. **void** printByte(string str)
35. {
36. **int** i=0;
37. **for** (i=0; i<str.length(); i++)
38. {
39. printf("%X ",(unsigned **char**)str.at(i));
40. }
41. printf("\n");
42. }
44. **void** wprintByte(wstring str)
45. {
46. **int** i=0;
47. **for** (i=0; i<str.length()\***sizeof**(**wchar\_t**); i++)
48. {
49. printf("%X ",\*((unsigned **char**\*)str.c\_str()+i));
50. }
51. printf("\n");
52. }
54. **int** main()
55. {
56. string strText = "AB汉字";
57. string strUTF8;
58. wstring wstrUnicode;
59. string strGB2312;
61. gb2312ToUnicode(strText, wstrUnicode);
62. printf("Unicode=");
63. wprintByte(wstrUnicode);
65. unicodeToUTF8(wstrUnicode, strUTF8);
66. printf("UTF-8  =");
67. printByte(strUTF8);
69. utf8ToUnicode(strUTF8,wstrUnicode);
70. printf("Unicode=");
71. wprintByte(wstrUnicode);
73. unicodeToGB2312(wstrUnicode,strGB2312);
74. printf("GB2312 =");
75. printByte(strGB2312);
77. **return** 0;
78. }

这里用“AB汉字”这样一个字符串做测试，它的ASCII编码为41 42 BA BA D7 D6  
输出结果：



转自：<http://www.cnweblog.com/fly2700/archive/2010/05/10/311354.html>