

常用数据类型使用转换详解计算机二级考试 PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/598/2021_2022__E5_B8_B8_E7_94_A8_E6_95_B0_E6_c97_598185.htm 特别推荐：2009年9月全国计算机等级考试时间及科目预告 2009年NCRE考试有新变化 2009年全国计算机等级考试大纲 2009年上半年全国计算机二级考试试题及答案 2009年上半年全国计算机等级考试试题答案汇总 本文将介绍一些常用数据类型的使用。我们先定义一些常见类型变量借以说明 int i = 100. long l = 2001. float f=300.2. double d=12345.119. char username[]="程佩君". char temp[200]. char *buf. CString str. _variant_t v1. _bstr_t v2. 一、其它数据类型转换为字符串 短整型(int) itoa(i,temp,10).///将i转换为字符串放入temp中,最后一个数字表示十进制 itoa(i,temp,2). ///按二进制方式转换 长整型(long) ltoa(l,temp,10). 浮点数(float,double) 用fcvt可以完成转换,这是MSDN中的例子: int decimal, sign. char *buffer. double source = 3.1415926535. buffer = _fcvt(source, 7, amp.sign). 运行结果:source: 3.1415926535 buffer: 31415927 decimal: 1 sign: 0 decimal表示小数点的位置,sign表示符号:0为正数, 1为负数 CString变量 str = "2008北京奥运". buf = (LPSTR)(LPCTSTR)str. BSTR变量 BSTR bstrValue = ::SysAllocString(L"程序员"). char * buf = _com_util::ConvertBSTRToString(bstrValue). SysFreeString(bstrValue). AfxMessageBox(buf). 0delete(buf). CComBSTR变量 CComBSTR bstrVar("test"). char *buf = _com_util::ConvertBSTRToString(bstrVar.m_str). AfxMessageBox(buf). 0delete(buf). _bstr_t变量 _bstr_t类型是

对BSTR的封装，因为已经重载了=操作符，所以很容易使用

```
_bstr_t bstrVar("test"). const char *buf = bstrVar.///不要修改buf中的内容 AfxMessageBox(buf). 通用方法(针对非COM数据类型)用sprintf完成转换 char buffer[200]. char c = 1. int i = 35. long j = 1000. float f = 1.7320534f. sprintf( buffer, "%c",c). sprintf( buffer, "%d",i). sprintf( buffer, "%d",j). sprintf( buffer, "%f",f). 二、字符串转换为其它数据类型 strcpy(temp,"123"). 短整型(int) i = atoi(temp). 长整型(long) l = atol(temp). 浮点(double) d = atof(temp). CString变量 CString name = temp. BSTR变量 BSTR bstrValue = ::SysAllocString(L"程序员"). ...///完成对bstrValue的使用 SysFreeString(bstrValue). CComBSTR变量 CComBSTR类型变量可以直接赋值 CComBSTR bstrVar1("test"). CComBSTR bstrVar2(temp). _bstr_t变量 _bstr_t类型的变量可以直接赋值 _bstr_t bstrVar1("test"). _bstr_t bstrVar2(temp). 三、其它数据类型转换到CString 使用CString的成员函数Format来转换,例如: 整数(int) str.Format("%d",i). 浮点数(float) str.Format("%f",i). 字符串指针(char *)等已经被CString构造函数支持的数据类型可以直接赋值 str = username. 对于Format所不支持的数据类型，可以通过上面所说的关于其它数据类型转化到char *的方法先转到char *，然后赋值给CString变量。 四、BSTR、_bstr_t与CComBSTR CComBSTR 是ATL对BSTR的封装，_bstr_t是C对BSTR的封装,BSTR是32位指针,但并不直接指向字串的缓冲区。 char *转换到BSTR可以这样: BSTR b=_com_util::ConvertStringToBSTR("数据").///使用前需要加上comutil.h和comsupp.lib SysFreeString(bstrValue). 反之可以使用 char *p=_com_util::ConvertBSTRToString(b). 0delete p. 具体
```

可以参考一，二段落里的具体说明。 CComBSTR与_bstr_t对大量的操作符进行了重载，可以直接进行=,!=,==等操作，所以使用非常方便。特别是_bstr_t,建议大家使用它。 五

、 VARIANT、_variant_t与 COleVariant VARIANT的结构可以参考头文件VC98\Include\OAIDL.H中关于结构体tagVARIANT的定义。对于VARIANT变量的赋值：首先给vt成员赋值，指明数据类型，再对联合结构中相同数据类型的变量赋值，举个例子： VARIANT va. int a=2001. va.vt=VT_I4.///指明整型数据 va.lVal=a. ///赋值 对于不马上赋值的VARIANT，最好先用Void VariantInit(VARIANTARG FAR* pvarg).进行初始化,其本质是将vt设置为VT_EMPTY,下表我们列举vt与常用数据的对应关系:

| | | | | | | | | | |
|-----------------------|-------------------------|------------------------|--------------------|--------------------------|----------------------|------------------------|--------------------|-----------------------------|----------------------|
| Byte bVal. | // VT_UI1. | Short iVal. | // VT_I2. | long lVal. | // VT_I4. | floatfltVal. | // VT_R4. | double dblVal. | // VT_R8. |
| VARIANT_BOOL boolVal. | // VT_BOOL. | SCODE scode. | // VT_ERROR. | CY cyVal. | // VT_CY. | DATE date. | // VT_DATE. | BSTR bstrVal. | // VT_BSTR. |
| DECIMAL FAR* pdecVal | // VT_BYREF VT_DECIMAL. | IUnknown FAR* punkVal. | // VT_UNKNOWN. | IDispatch FAR* pdispVal. | // VT_DISPATCH. | SAFEARRAY FAR* parray. | // VT_ARRAY *. | Byte FAR* pbVal. | // VT_BYREF VT_UI1. |
| short FAR* piVal. | // VT_BYREF VT_I2. | long FAR* plVal. | // VT_BYREF VT_I4. | float FAR* pfltVal. | // VT_BYREF VT_R4. | double FAR* pdblVal. | // VT_BYREF VT_R8. | VARIANT_BOOL FAR* pboolVal. | // VT_BYREF VT_BOOL. |
| SCODE FAR* pscode. | // VT_BYREF VT_ERROR. | CY FAR* pcyVal. | // VT_BYREF VT_CY. | DATE FAR* pdate. | // VT_BYREF VT_DATE. | BSTR FAR* pbstrVal. | // | | |

```

VT_BYREF|VT_BSTR. IUnknown FAR* FAR* ppunkVal. //
VT_BYREF|VT_UNKNOWN. IDispatch FAR* FAR* ppdispVal.
// VT_BYREF|VT_DISPATCH. SAFEARRAY FAR* FAR*
pparray. // VT_ARRAY|*. VARIANT FAR* pvarVal. //
VT_BYREF|VT_VARIANT. void FAR* byref. // Generic ByRef.
char cVal. // VT_I1. unsigned short uiVal. // VT_UI2. unsigned
long ulVal. // VT_UI4. int intVal. // VT_INT. unsigned int uintVal.
// VT_UINT. char FAR * pcVal. // VT_BYREF|VT_I1. unsigned
short FAR * puiVal. // VT_BYREF|VT_UI2. unsigned long FAR *
pulVal. // VT_BYREF|VT_UI4. int FAR * pintVal. //
VT_BYREF|VT_INT. unsigned int FAR * pointVal.
//VT_BYREF|VT_UINT. _variant_t是VARIANT的封装类，其
赋值可以使用强制类型转换，其构造函数会自动处理这些数
据类型。使用时需加上#include <...>. 例如：long l=222. int i=100.
_variant_t IVal(l). IVal = (long)i. COleVariant的使用与_variant_t
的方法基本一样，请参考如下例子：COleVariant v3 = "字符串", v4 = (long)1999. CString str =(BSTR)v3.pbstrVal. long i =
v4.IVal. 六、其它一些COM数据类型 根据ProgID得到CLSID
HRESULT CLSIDFromProgID( LPCOLESTR
lpszProgID,LPCLSID pclsid). CLSID clsid. CLSIDFromProgID(
L"MAPI.Folder",amp.pProgID). ...///可以使用pProgID
CoTaskMemFree(pProgID).//不要忘记释放 七、ANSI
与Unicode Unicode称为宽字符型字串,COM里使用的都
是Unicode字符串。将ANSI转换到Unicode (1)通过L这个宏来
实现，例如: CLSIDFromProgID( L"MAPI.Folder",amp.clsid).
将Unicode转换到ANSI (1)使用WideCharToMultiByte,例如: //

```

假设已经有了一个Unicode 串 wszSomeString... char szANSIString [MAX_PATH]. WideCharToMultiByte (CP_ACP, WC_COMPOSITECHECK, wszSomeString, -1, szANSIString, sizeof(szANSIString), NULL, NULL). (2)使用W2A宏来实现,例如: USES_CONVERSION. pTemp=W2A(wszSomeString). 八、其它对消息的处理中我们经常需要将WPARAM或LPARAM等32位数据 (DWORD)分解成两个16位数据 (WORD),例如 : LPARAM lParam. WORD loValue = LOWORD(lParam).///取低16位 WORD hiValue = HIWORD(lParam).///取高16位 对于16位的数据(WORD)我们可以用同样的方法分解成高低两个8位数据(BYTE),例如: WORD wValue. BYTE loValue = LOBYTE(wValue).///取低8位 BYTE hiValue = HIBYTE(wValue).///取高8位 两个16位数据 (WORD) 合成32位数据(DWORD,LRESULT,LPARAM,或WPARAM) LONG MAKELONG(WORD wLow, WORD wHigh). WPARAM MAKEWPARAM(WORD wLow, WORD wHigh). LPARAM MAKELPARAM(WORD wLow, WORD wHigh). LRESULT MAKELRESULT(WORD wLow, WORD wHigh). 两个8位的数据(BYTE)合成16位的数据(WORD) WORD MAKEWORD(BYTE bLow, BYTE bHigh). 从R(red),G(green),B(blue)三色得到COLORREF类型的颜色值 COLORREF RGB(BYTE byRed,BYTE byGreen,BYTE byBlue). 例如COLORREF bkcolor = RGB(0x22,0x98,0x34). 从COLORREF类型的颜色值得到RGB三个颜色值 BYTE Red = GetRValue(bkcolor). ///得到红颜色 BYTE Green = GetGValue(bkcolor). ///得到绿颜色 BYTE Blue = GetBValue(bkcolor). ///得到兰颜色 九、注意事项 假如需要使

用到ConvertBSTRToString此类函数,需要加上头文件comutil.h,并在setting中加入comsupp.lib或者直接加上#pragma comment(lib, "comsupp.lib")

1. CString与LPCTSTR(const wchar_t *)
CString转LPCTSTR: CString str. LPCTSTR lpstr = str.GetBuffer().

LPCTSTR转CString: LPCTSTR lpstr. CString str(lpstr). 2. BSTR

与char *(来自互联网) BSTR转char*: char* lpszText = "Test".

printf("char * text: %s\n", lpszText). BSTR bstrText =

_com_util::ConvertStringToBSTR(lpszText). wprintf(L"BSTR text: %s\n", bstrText). SysFreeString(bstrText).

char*转BSTR: BSTR bstrText = ::SysAllocString(L"Test"). wprintf(L"BSTR text: %s\n", bstrText).

char* lpszText2 =

_com_util::ConvertBSTRToString(bstrText). printf("char * text: %s\n", lpszText2). SysFreeString(bstrText). 0delete[] lpszText2.

编辑特别推荐: 全国计算机等级考试 (等考) 指定教材 全国计算机等级考试学习视频 全国计算机等级考试网上辅导招生 全国计算机等级考试时间及科目预告 百考试题教育全国计算机等级考试在线测试平台 全国计算机等级考试资料下载 全国计算机等级考试论坛 100Test 下载频道开通, 各类考试题目直接下载。详细请访问 www.100test.com