# 这一节我们来学习获取磁盘容量以及可用空间和其他信息

# 学习大纲

|  |
| --- |
|  |

## GetDiskFreeSpace函数的语法如下

|  |
| --- |
| 语法 C++复制  BOOL GetDiskFreeSpaceA(  [in] LPCSTR lpRootPathName,  [out] LPDWORD lpSectorsPerCluster,  [out] LPDWORD lpBytesPerSector,  [out] LPDWORD lpNumberOfFreeClusters,  [out] LPDWORD lpTotalNumberOfClusters  ); 参数 [in] lpRootPathName  要返回信息的磁盘的根目录。如果该参数为 **NULL，**则函数使用当前磁盘的根目录。如果此参数是 UNC 名称，则 它必须包含尾部反斜杠（例如，“\\MyServer\MyShare\”）。此外，驱动器 规范必须具有尾部反斜杠（例如，“C：\”）。调用应用程序必须 具有此目录的**FILE\_LIST\_DIRECTORY**访问权限。  [out] lpSectorsPerCluster  一个指向变量的指针，该变量接收每个集群的扇区数。  [out] lpBytesPerSector  指向接收每个扇区字节数的变量的指针。  [out] lpNumberOfFreeClusters  一个指向变量的指针，该变量接收磁盘上可用于 与调用线程关联的用户。  如果正在使用每用户磁盘配额，则此值可能小于 磁盘。  [out] lpTotalNumberOfClusters  指向一个变量的指针，该变量接收磁盘上可供用户使用的群集总数 谁与调用线程相关联。  如果正在使用每用户磁盘配额，则此值可能小于磁盘上的群集总数。 返回值 如果函数成功，则返回值为非零。  如果函数失败，则返回值为零。要获取扩展错误信息，请调用 [GetLastError](https://learn.microsoft.com/en-us/windows/desktop/api/errhandlingapi/nf-errhandlingapi-getlasterror)。 言论 [GetDiskFreeSpaceEx](https://learn.microsoft.com/en-us/windows/desktop/api/fileapi/nf-fileapi-getdiskfreespaceexa) 函数允许 您可以避免 **GetDiskFreeSpace** 函数所需的一些算术运算。 |

## GetDiskFreeSpaceEx函数的语法如下

|  |
| --- |
| 语法 C++  BOOL GetDiskFreeSpaceExA(  [in, optional] LPCSTR lpDirectoryName,  [out, optional] PULARGE\_INTEGER lpFreeBytesAvailableToCaller,  [out, optional] PULARGE\_INTEGER lpTotalNumberOfBytes,  [out, optional] PULARGE\_INTEGER lpTotalNumberOfFreeBytes  ); 参数 [in, optional] lpDirectoryName  磁盘上的目录。  如果该参数为 **NULL，**则函数使用当前磁盘的根目录。  如果此参数是 UNC 名称，则它必须包含尾部反斜杠，例如 “\\MyServer\MyShare\”。  此参数不必指定磁盘上的根目录。该函数接受 磁盘。  调用应用程序必须具有此目录的**FILE\_LIST\_DIRECTORY**访问权限。  [out, optional] lpFreeBytesAvailableToCaller  指向一个变量的指针，该变量接收磁盘上可供用户使用的可用字节总数 与调用线程相关联。  此参数可以为 **NULL。**  如果使用每用户配额，则此值可能小于磁盘上的可用字节总数。  [out, optional] lpTotalNumberOfBytes  一个指向变量的指针，该变量接收磁盘上可供用户使用的总字节数 与调用线程相关联。  此参数可以为 **NULL。**  如果使用每用户配额，则此值可能小于磁盘上的字节总数。  要确定磁盘或卷上的字节总数，请使用 [IOCTL\_DISK\_GET\_LENGTH\_INFO](https://learn.microsoft.com/en-us/windows/desktop/api/winioctl/ni-winioctl-ioctl_disk_get_length_info)。  [out, optional] lpTotalNumberOfFreeBytes  指向接收磁盘上可用字节总数的变量的指针。  此参数可以为 **NULL。** 返回值 如果函数成功，则返回值为非零。  如果函数失败，则返回值为零 （0）。要获取扩展错误信息，请调用 [GetLastError](https://learn.microsoft.com/en-us/windows/desktop/api/errhandlingapi/nf-errhandlingapi-getlasterror)。 言论 此函数获取的值为 [ULARGE\_INTEGER](https://learn.microsoft.com/en-us/windows/win32/api/winnt/ns-winnt-ularge_integer~r1) 类型。不要将这些值截断为 32 位。  **GetDiskFreeSpaceEx** 函数返回零 （0） 对于 *lpTotalNumberOfFreeBytes* 和 *lpFreeBytesAvailable* for all CD 请求，除非该磁盘是 CD-RW 驱动器中未写入的 CD。 |

### 其实GetDiskFreeSpaceEx函数获取可用磁盘空间比较方便，注意，他的参数类PULARGE\_INTEGER是指向一个有两个结构体组成 的联合体的指针，这个ULARGE\_INTEGER的QuadPart属性才是数字，需要连续3次除以1024才能变为GB，可用磁盘空间是指lpTotalNumberOfFreeBytes

### GetDiskFreeSpace可用获取到总簇数cluster，每一个簇的扇区数以及每个扇区的字节数，需要把这首歌数相乘，就可以得到磁盘空间需要连续3次除以1024才能变为GB，用GetDiskFreeSpaceEx比较方便。

# 演练

## 1.新建一个常规空项目，取名Lesson14-get-disk-space,然后新建一个cpp文件取名：disk-space.cpp，代码如下

|  |
| --- |
| #include<Windows.h>  #include<stdio.h>  #include<stdlib.h>  #define BUFSIZE 1024  BOOL testGetDiskFreeSpace(LPSTR szDrive);  BOOL testGetDiskFreeSpaceEx(LPSTR szDrive);  int main()  {  CHAR szLogicalStr[BUFSIZE];  PCHAR szDrive;  ZeroMemory(szLogicalStr,BUFSIZE);  GetLogicalDriveStrings(BUFSIZE-1,szLogicalStr);  szDrive = (PCHAR)szLogicalStr;  do  {  printf("Drive: %s\n",szDrive);  /\* if(!testGetDiskFreeSpace(szDrive))  {  printf("test Failed...\n");  return 1;  }\*/  if(!testGetDiskFreeSpaceEx(szDrive))  {  printf("test Failed...\n");  return 1;  }  szDrive +=(lstrlen(szDrive)+1);  printf("===========\n");  } while(lstrlen(szDrive)!='\x00');  system("pause");  return 0;  }  BOOL testGetDiskFreeSpace(LPSTR szDrive)  {  DWORD dwSectorsPerCluster; // pointer to sectors per cluster  DWORD dwBytesPerSector; // pointer to bytes per sector  DWORD dwNumberOfFreeClusters; // pointer to number of free clusters  DWORD dwTotalNumberOfClusters; // pointer to total number of clusters  BOOL bRet;  bRet = GetDiskFreeSpace(szDrive,&dwSectorsPerCluster,&dwBytesPerSector,  &dwNumberOfFreeClusters,&dwTotalNumberOfClusters);  if(!bRet)  {  printf("Get Disk Space Failed...");  }  printf("Basic Infos:\n");  printf("Sectors Per Cluster: %d\n",dwSectorsPerCluster);  printf("Bytes Per Sector: %d\n",dwBytesPerSector);  printf("Number Of Free Clusters: %d\n",dwNumberOfFreeClusters);  printf("Total Number Of Clusters: %d\n",dwTotalNumberOfClusters);  printf("Extented Infos:\n");  return bRet;  }  /\*\*  PULARGE\_INTEGER是指向一个有两个结构体组成的联合体的指针，  这个ULARGE\_INTEGER的QuadPart属性才是数字  \*/  BOOL testGetDiskFreeSpaceEx(LPSTR szDrive)  {  ULARGE\_INTEGER uFreeBytesAvailableToCaller; // receives the number of bytes ondisk available to the caller  ULARGE\_INTEGER uTotalNumberOfBytes; // receives the number of bytes on disk  ULARGE\_INTEGER uTotalNumberOfFreeBytes; // receives the free bytes on disk  BOOL bRet;  bRet = GetDiskFreeSpaceEx(szDrive,&uFreeBytesAvailableToCaller,  &uTotalNumberOfBytes,&uTotalNumberOfFreeBytes);  if(!bRet)  {  printf("Get Disk Space Failed...");  }  printf(" Free Bytes Available To Caller: %dGB\n", uFreeBytesAvailableToCaller.QuadPart/1024/1024/1024);  printf(" Total Number Of Bytes : %dGB\n", uTotalNumberOfBytes.QuadPart/1024/1024/1024);  printf(" Total Number Of FreeBytes: %dGB\n", uTotalNumberOfFreeBytes.QuadPart/1024/1024/1024);  return bRet;  } |

### testGetDiskFreeSpaceEx效果：

|  |
| --- |
|  |

### testGetDiskFreeSpaceEx效果需要计算使用这里就不演示了

## 这里有个宽字符版本

|  |
| --- |
| #include<Windows.h>  #include<stdio.h>  #include<stdlib.h>  #define BUFSIZE 1024  BOOL TestGetDiskFreeSPaceEx()  {  WCHAR driveStr[BUFSIZE];  memset(driveStr, 0, BUFSIZE);  WCHAR\* pDrive;  ULARGE\_INTEGER uFreeBytesAvailableToCaller;  ULARGE\_INTEGER uTotalNumberOfBytes;  ULARGE\_INTEGER uTotalNumberOfFreeBytes;  int factor = 1024 \* 1024 \* 1024;  //获取逻辑驱动器字符串  if (!GetLogicalDriveStrings(BUFSIZE - 1, driveStr))  {  wprintf\_s(L"Get Logical Drive String Failed,error:%d\n", GetLastError());  return FALSE;  }  pDrive = (WCHAR\*)driveStr;  do  {  wprintf\_s(L"Drive:%s\n", pDrive);  if (!GetDiskFreeSpaceEx(pDrive, &uFreeBytesAvailableToCaller, &uTotalNumberOfBytes, &uTotalNumberOfFreeBytes))  {  wprintf\_s(L"Get Logical Drive String Failed,error:%d\n", GetLastError());  return FALSE;  }  wprintf\_s(L"Free Bytes Available To Caller:%I64d GB\n", (\_\_int64)uFreeBytesAvailableToCaller.QuadPart / factor);  wprintf\_s(L"Total Number Of Bytes:%I64d GB\n", (\_\_int64)uTotalNumberOfBytes.QuadPart / factor);  wprintf\_s(L"Total Number Of Free Bytes:%I64d GB\n", (\_\_int64)uTotalNumberOfFreeBytes.QuadPart / factor);  pDrive += lstrlen(pDrive) + 1;  wprintf\_s(L"==========================================\n");  } while (\*pDrive != L'\x00');  return TRUE;  }  int main()  {  if (!TestGetDiskFreeSPaceEx())  {  wprintf\_s(L"Get Disk Free Space Failed,error:%d\n",GetLastError());  return -1;  }  system("pause");  return 0;  } |

# 这一节的学习到此为止