//查找指定文件路径的文件
  
HANDLE FindFirstFile(
  
 LPCTSTR lpFileName, //文件路径名
  
 LPWIN32\_FIND\_DATA lpFindFileData //指向一个用于保存文件信息的结构体
  
);
  
WIN32\_FIND\_DATA结构体如下:
  
typedef struct \_WIN32\_FIND\_DATA {
  
DWORD dwFileAttributes; // 文件属性
  
FILETIME ftCreationTime; // 文件创建时间
  
FILETIME ftLastAccessTime; // 文件最后一次访问时间
  
FILETIME ftLastWriteTime; // 文件最后一次修改时间
  
DWORD nFileSizeHigh; // 文件长度高 32 位
  
DWORD nFileSizeLow; // 文件长度低 32 位
  
DWORD dwReserved0; // 系统保留
  
DWORD dwReserved1; // 系统保留
  
TCHAR cFileName[MAX\_PATH]; // 长文件名
  
TCHAR cAlternateFileName[14]; // 8.3格式文件名
  
} WIN32\_FIND\_DATA, \*PWIN32\_FIND\_DATA;

//创建一个新的文件夹
  
BOOL CreateDirectory(
  
LPCTSTR lpPathName, //文件夹名称
  
LPSECURITY\_ATTRIBUTES lpSecurityAttributes //安全属性，一般设置为NULL即可
  
);

//查找下一个文件
  
BOOL FindNextFile(
  
HANDLE hFindFile, //搜索的文件句柄
  
LPWIN32\_FIND\_DATA lpFindFileData //指向一个用于保存文件信息的结构体
  
);

//打开或者创建一个文件
  
HANDLE CreateFile(
  
LPCTSTR lpFileName, //普通文件名或者设备文件名
  
DWORD dwDesiredAccess, //访问模式（写/读）
  
DWORD dwShareMode, //共享模式
  
LPSECURITY\_ATTRIBUTES lpSecurityAttributes, //指向安全属性的指针
  
DWORD dwCreationDisposition, //如何创建
  
DWORD dwFlagsAndAttributes, //文件属性
  
HANDLE hTemplateFile //用于复制文件句柄
  
);

BOOL SetFileTime(
  
 HANDLE hFile,
  
 const FILETIME \*lpCreationTime, //创建时间
  
 const FILETIME \*lpLastAccessTime, //最近访问时间
  
 const FILETIME \*lpLastWriteTime //最后修改时间
  
);

BOOL ReadFile(
  
 HANDLE hFile,
  
 LPVOID lpBuffer,
  
 DWORD nNumberOfBytesToRead,
  
 LPDWORD lpNumberOfBytesRead, //指向实际读取字节数的指针
  
 LPOVERLAPPED lpOverlapped
  
);

BOOL WriteFile(
  
HANDLE hFile, //文件句柄
  
LPCVOID lpBuffer, //数据缓存区指针
  
DWORD nNumberOfBytesToWrite, //要写的字节数
  
LPDWORD lpNumberOfBytesWritten, //用于保存实际写入字节数的存储区域的指针
  
LPOVERLAPPED lpOverlapped //OVERLAPPED 结构体指针
  
);