

Bài tập. Xây dựng ứng dụng web server quản lý thư mục / tập tin

Yêu cầu: Lập trình tạo một web server trả về cấu trúc thư mục của ổ đĩa trên server. Tùy thuộc vào trình duyệt gửi yêu cầu nội dung của thư mục hay của tập tin mà server sẽ trả về dữ liệu thích hợp.

Sử dụng hàm Windows API để duyệt danh sách thư mục con / tập tin của 1 thư mục

- **Mục đích:** Sử dụng hàm **FindFirstFileA()** và **FindNextFileA()** để lấy danh sách thư mục và tập tin của một thư mục bất kỳ.
- Cấu trúc **WIN32_FIND_DATA** được sử dụng để lưu dữ liệu trả về bởi 2 hàm trên.

```
typedef struct _WIN32_FIND_DATA {  
    DWORD        dwFileAttributes;           // Thuộc tính file  
    FILETIME     ftCreationTime;  
    FILETIME     ftLastAccessTime;  
    FILETIME     ftLastWriteTime;  
    DWORD        nFileSizeHigh;              // Kích thước file  
    DWORD        nFileSizeLow;              // Kích thước file  
    DWORD        dwReserved0;  
    DWORD        dwReserved1;  
    TCHAR        cFileName[MAX_PATH];       // Tên file  
    TCHAR        cAlternateFileName[14];  
} WIN32_FIND_DATA, *PWIN32_FIND_DATA, *LPWIN32_FIND_DATA;
```

- Các trường cần quan tâm gồm **dwFileAttributes** chứa thuộc tính của thư mục/tập tin, được sử dụng để phân biệt thư mục hoặc tập tin, **cFileName** chứa tên của thư mục/tập tin, **nFileSizeHigh** và **nFileSizeLow** chứa kích thước của tập tin.
- Kích thước của file được xác định theo công thức:

$$(\mathbf{nFileSizeHigh * (MAXDWORD + 1)}) + \mathbf{nFileSizeLow}$$

- Đoạn mã sau được sử dụng để duyệt tất cả các thư mục và tập tin trong thư mục C:

```
WIN32_FIND_DATA DATA;  
HANDLE h = FindFirstFileA("C:\\*.\"", &DATA);  
do {
```

```
if (DATA.dwFileAttributes & FILE_ATTRIBUTE_DIRECTORY) {  
    // Đây là một thư mục  
    // In ra tên thư mục  
} else {  
    // Đây là một file  
    // In ra tên file và kích thước file  
}  
} while (FindNextFileA(h, &DATA));
```