# 71. Simplify Path

Given an absolute path for a Unix-style file system, which begins with a slash '/', transform this path into its simplified canonical path.

In Unix-style file system context, a single period '.' signifies the current directory, a double period ".." denotes moving up one directory level, and multiple slashes such as "//" are interpreted as a single slash. In this problem, treat sequences of periods not covered by the previous rules (like "...") as valid names for files or directories.

The simplified canonical path should adhere to the following rules:

It must start with a single slash '/'.

Directories within the path should be separated by only one slash '/'.

It should not end with a slash '/', unless it's the root directory.

It should exclude any single or double periods used to denote current or parent directories.

Return the new path.

## Solution in C++

class Solution {

public:

string simplifyPath(string path) {

vector<string> lks;

for (int i = 0; i < path.length();) {

if (path[i] != '/') {

string dir = "";

while (i < path.length() && path[i] != '/') dir+=path[i++];

if(dir == ".."){

if(lks.size())

lks.pop\_back();

}

else if (dir != ".") {

lks.push\_back(dir);

}

}

else ++i;

}

string ans="";

for(string x : lks) ans += "/"+ x;

return !ans.length() ? "/" :ans;

}

};