**In Memory File System**

**Problem**

Implement the basic \*nix file system operations, such as ls, mkdir -p, add file, etc., of an in-memory file system. An in-memory filesystem only exist in memory and is not persisted on disk.

**Instructions**

Please implement Ls, MkdirP, AddFileWithContent, and GetFileContent in FileSystem class. For the sake of simplicity, you can assume all file or directory paths are absolute paths which begin with / and do not end with /. You can also assume that all operations will be passed valid parameters and users will not attempt to retrieve file content or list a directory that does not exist.

Please do not implement using local filesystem, such as the DIR, fstream and iostream.

For example,

[1](https://www.oneinterview.io/interviews/0391482579" \l "n1)// assumption: all path starts with / and not ending with /

[2](https://www.oneinterview.io/interviews/0391482579" \l "n2)FileSystem fs = FileSystem();

[3](https://www.oneinterview.io/interviews/0391482579" \l "n3)

[4](https://www.oneinterview.io/interviews/0391482579" \l "n4)// should print []

[5](https://www.oneinterview.io/interviews/0391482579" \l "n5)cout << fs.Ls("/") << endl;

[6](https://www.oneinterview.io/interviews/0391482579" \l "n6)fs.MkdirP("/a/b/c");

[7](https://www.oneinterview.io/interviews/0391482579" \l "n7)fs.AddFileWithContent("/a/b/c/d", "hello world");

[8](https://www.oneinterview.io/interviews/0391482579" \l "n8)// should print [a]

[9](https://www.oneinterview.io/interviews/0391482579" \l "n9)cout << fs.Ls("/") << endl;

**[10](https://www.oneinterview.io/interviews/0391482579" \l "n10)**// should print [d]

[11](https://www.oneinterview.io/interviews/0391482579" \l "n11)cout << fs.Ls("/a/b/c") << endl;

[12](https://www.oneinterview.io/interviews/0391482579" \l "n12)// should print [d]

[13](https://www.oneinterview.io/interviews/0391482579" \l "n13)cout << fs.Ls("/a/b/c/d") << endl;

[14](https://www.oneinterview.io/interviews/0391482579" \l "n14)// should print hello world

[15](https://www.oneinterview.io/interviews/0391482579" \l "n15)cout << fs.GetFileContent("/a/b/c/d") << endl;

#include <iostream>

#include <map>

#include <sstream>

#include <string>

#include <vector>

using namespace std;

vector<string> split(const string &s, char delim) {

vector<string> elems;

stringstream ss(s);

string item;

while (getline(ss, item, delim)) {

elems.push\_back(item);

}

return elems;

}

string join(const vector<string>& input) {

string output = "";

bool has\_data = false;

for (size\_t i = 0; i < input.size(); ++i) {

has\_data = true;

output += input[i];

output += ", ";

}

if (has\_data) {

output = output.substr(0, output.length() - 2);

}

return output;

}

class FileSystem {

public:

vector<string> Ls(const string& path) {

// IMPLEMENT ME

vector<string> output;

return output;

}

void MkdirP(const string& dir\_path) {

// IMPLEMENT ME

}

void AddFileWithContent(const string& file\_path, const string& content) {

// IMPLEMENT ME

}

string GetFileContent(const string& file\_path) {

// IMPLEMENT ME

return "";

}

};

#ifndef \_\_main\_\_

#define \_\_main\_\_

int main(int argc, char\* argv[]) {

FileSystem fs;

// should print ""

cout << join(fs.Ls("/")) << endl;

fs.MkdirP("/a/b/c");

fs.AddFileWithContent("/a/b/c/d", "hello world");

// should print a

cout << join(fs.Ls("/")) << endl;

// should print d

cout << join(fs.Ls("/a/b/c")) << endl;

// should print d

cout << join(fs.Ls("/a/b/c/d")) << endl;

// should print hello world

cout << fs.GetFileContent("/a/b/c/d") << endl;

}

#endif