/\*

\* File: main.cpp

\* Author: Ryanfinazzo

\*

\* Created on May 31, 2018, 3:32 PM

\*/

#include <cstdlib>

#include <iostream>

#include <cstring>

#include <string>

#include <sstream>

using namespace std;

class File{

private:

string name;

string data;

public:

File(){

}

File(string name ){

setName(name);

}

void print(){

cout<<"\t"<<getName()<<endl;;

}

void setData(string data){

this->data =data;

}

string getData(){

return this->data;

}

void setName(string name){

this->name =name;

}

string getName(){

return this->name;

}

};

class Directory{

private:

string name;

File files [100];

int filecount=0;

public:

Directory(){

}

Directory(string name){

setName(name);

}

void addf(File f){

files[filecount]=f;

filecount++;

}

void setName(string name){

this->name =name;

}

string getName(){

return this->name;

}

string getfileName(int j){

return files[j].getName();

}

int getfileCount(){

return filecount;

}

void delfile(int pos, int fc){

pos--;

for(int i=pos; i<=fc; i++){

files[i]=files[i+1];

}

}

void print(){

cout<<this->getName()<<endl;

for (int i=0;i<filecount; i++){

files[i].print();

}

}

string getfileData(int j){

return files[j].getData();

}

void setfileData(string data,int j){

files[j].setData(data);

}

};

class FileSystem{

private:

Directory dir[100];

public:

FileSystem(){

}

void add (Directory d, int i){

dir[i]=d;

}

void addfile(File f, int i){

dir[i].addf(f);

}

void print(int dircount){

for (int i=0;i<dircount; i++){

dir[i].print();

}

}

string getdirName(int i){

return dir[i].getName();

}

string getfileName(int i,int j){

return dir[i].getfileName(j);

}

int getfileCount(int i){

return dir[i].getfileCount();

}

void printfileData(int i,int j){

cout<<dir[i].getfileData(j);

}

void setfileData(string data, int i, int j){

dir[i].setfileData(data,j);

}

void deldir(int pos, int dircount){

pos--;

for(int i =pos; i<=dircount; i++){

dir[i]=dir[i+1];

}

}

void delfile(int i, int j, int fc){

dir[i].delfile(j, fc);

}

};

int main(int ac, char\* args[]) {

FileSystem filesys;

int dircount=0;

string input [4];

do{

cout<<"Valid input is as follows:"<<endl;

cout<<"--------------------------"<<endl;

cout<<"CREATE directory name"<<endl;

cout<<"CREATE file name dirname"<<endl;

cout<<"DELETE directory name"<<endl;

cout<<"DELETE file name"<<endl;

cout<<"READ name"<<endl;

cout<<"WRITE filename data"<<endl;

cout<<"EXIT"<<endl;

cout<<"---------------------------"<<endl;

cout<<"type = 'directory' or 'file'"<<endl;

cout<<"name = name of directory or file"<<endl;

cout<<"filename = name of file to write data in"<<endl;

cout<<"dirname = name of directory to put file in"<<endl<<endl;

cout<<"Enter a Command"<<endl;

string s;

getline(cin, s);

string delimiter = " ";

size\_t pos = 0;

int i=0;

while ((pos = s.find(delimiter)) != string::npos) {

input[i] = s.substr(0, pos);

s.erase(0, pos + delimiter.length());

i++;

}

input[i++]=s;

if (input[0]=="EXIT"){

return 0;

}

else if (input[0]=="READ"){

int temp =0;

for (int i=0;i<dircount;i++){

int temp=filesys.getfileCount(i);

for(int j=0;j<temp;j++){

if (args[1]==(filesys.getfileName(i,j))){

filesys.printfileData(i,j);

temp =1;

break;

}

}

if (temp){

break;

}

}

}

else if (input[0]=="DELETE"){

if (input[1]=="directory"){

for (int i = 0;i<dircount;i++){

if (input[2]==(filesys.getdirName(i))){

filesys.deldir(i, dircount);

break;

}

}

}

else if (input[1]=="file"){

int temp3=0;

for (int i=0;i<dircount;i++){

int fc=filesys.getfileCount(i);

for(int j=0;j<fc;j++){

if (input[2]==(filesys.getfileName(i,j))){

filesys.delfile(i,j, fc);

temp3=1;

break;

}

}

if (temp3){

break;

}

}

}

}

else if (input[0]=="CREATE"){

if (input[1]=="directory"){

Directory d(input[2]);

filesys.add(d, dircount);

dircount++;

}

else if (input[1]=="file"){

for (int i=0;i<dircount;i++){

string dname = filesys.getdirName(i);

if (dname==input[3]){

File f(input[2]);

filesys.addfile(f, i);

}

}

}

}

else if (input[0]=="WRITE"){

int temp=0;

for (int i=0;i<dircount;i++){

int fc=filesys.getfileCount(i);

for(int j=0;j<temp;j++){

if (input[1]==(filesys.getfileName(i,j))){

filesys.setfileData(input[2],i,j);

temp=1;

break;

}

}

if (temp){

break;

}

}

}

cout<<endl<<endl<<"CURRENT FILE SYSTEM"<<endl;

cout<<"------------------"<<endl;

filesys.print(dircount);

cout<<endl<<endl;

}while(input[0]!="EXIT");

return 0;

}