National University of Computer & Emerging Sciences (NUCES),
Islamabad Department of Software Engineering

CS-2001 Data Structures (Fall 2023) Assignment #03

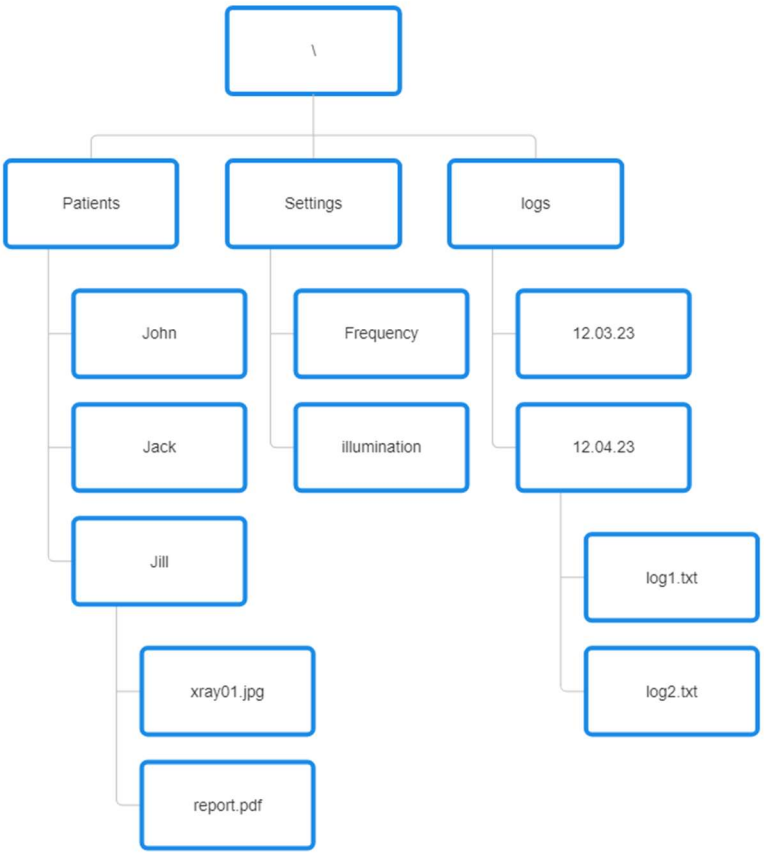
Developing a File Directory Tree Shell for Medical Device Embedded System

Design and implement a File Directory Tree Shell tailored for a small, embedded operating system used in medical devices. This shell will enable the management of files and directories, with a focus on efficiency and resource optimization.

Task Description:

You have been enlisted by a company specializing in medical device embedded systems. They are developing a new medical device that necessitates a compact operating system for debugging purposes. This operating system will exclusively manage files categorized as patient data, device settings, and logs. Your mission is to create a File Directory Tree Shell customized for their specific needs.

NOTE: YOU MAY RESTIRCT THIS TO BINARY TREE AND MAINTAIN ONLY PATIENT DATA AND LOGS.



I. File Directory Structure (15 Marks)

1. Implement a tree data structure to represent files and directories within the operating system. Each node in the tree should have the following attributes:

- Name: The name of the file or directory.
- Path: The full path to the file or directory.
- Type: Indication of whether the node is a directory or file.
- Children: A list of child nodes.

2. Develop a function to create a new directory within the tree. Users should be able to specify the directory name.

3. Create a function to create new files within the tree, allowing users to specify the name, type (e.g., txt, pdf), and complete path.

II. Core Functionality (25 Marks)

4. Implement functionalities for:

- Listing the file directory tree in a tree view (level order).
- Merging directories, combining the contents of a given subtree or tree.
- Deleting and renaming files and directories.
- Searching for files and directories by name or content.
- Copying and moving files and directories between different locations.
- Exporting and importing the file directory tree to and from a file.

III. Error Handling (10 Marks)

5. Implement comprehensive error handling for common scenarios, including but not limited to attempting to delete non-empty directories, moving to non-existent locations, and other potential pitfalls. During merging or moving, check for duplicates as well.

Deliverables:

Submit the C++ source code for the File Directory Tree Shell. Develop a user-friendly command-line interface (CLI) for interacting with the shell, focusing on simplicity and clarity.

Assessment Criteria:

- Proper implementation of the tree data structure (15 Marks)
- Accurate implementation of core functionalities (25 Marks)
- Effective error handling (10 Marks)

Instructions (before starting the assignment):

1. Assignments are to be done individually.

2. The code you write must be your own and you must understand each part of your code. You are encouraged to get help from the course instructors through google classroom and email.

3. Apply all validations for invalid inputs.

4. Plagiarism: Plagiarism of any kind (copying from others, copying from the internet, etc.) is not allowed. If found plagiarized, you will be awarded zero marks in the assignment. Repeating such an act can lead to strict disciplinary actions and failure in the course.

5. Please start early otherwise you will struggle with the assignment.

Submission Guidelines

a. submit both .h and .cpp file Your submission must contain your name, student-id, and assignment # on the top of the file in the comments. Example the first line of assignment you should write //Maheen_Arshad_22i111. Missing this will result in 20% marks deduction in each question.

b. Move your .cpp and .h file in one folder. The folder must contain only submission.cpp and h files(no binaries, no exe files etc.,). If we are unable to download your submission due to any reason you will be awarded zero mark.

c. Run and test your program on machine before submission. If there is a syntax error, zero marks will be awarded in that specific question.

d. Rename the folder as ROLL-NUM_SECTION (e.g. 21i-0001_A) and compress the folder as a **zip file**. (e.g. 21i-0001_A.zip). Only **zip file** will be acceptable.

e. Submit the .zip file on Google Classroom within the deadline. Late submission will be marked zero. No exceptions

f. Submission other than Google classroom (e.g. email etc.) will not be accepted.

g. The student is solely responsible to check the final zip files for issues like corrupt files, viruses in the file, mistakenly exe sent. If we cannot download the file from Google classroom due to any reason it will lead to zero marks in the assignment.