Due: Feb 24, Wednesday

In this assignment, you are asked to use a tree that can handle unknown number of children to represent the file system of Unix. Let the root of the tree represent the root directory. Each internal node represents a directory where the contents (files and directories under the directory) are stored as children. The leaves of the tree will be files or empty directories. You should use the tree structure introduced in the class for this assignment (or study Section 4.1.)

1. Make directory ∼/IT279/Asg2/ on your Unix account for this assignment, where ∼ denotes your home directory.

2. Use the following Unix command to create your own directory file, `allUsersDir.txt`

   ```
   dir -R /home/ > ∼/IT279/Asg2/allUsersDir.txt
   ```

   You will see some permission denied messages on the screen, but these error message will not be written into `allUsersDir.txt`. Different student will have different results. So, don't share this file.

3. Check the contents of `allUsersDir.txt`. This file contains all users and the contents of users' directories, if you have the permission. Two different directories are separated by an empty line. Each directory starts with a line that contains the directory's full path followed by a colon ":". The second line indicates total space used for this directory. Starting from the third line, each line is an entry of the directory. We are interested in the 9th column, file of directory names.

4. Write a C++ program named `dir.cpp` that always reads in your `allUsersDir.txt` and build a directory tree based on your `allUsersDir.txt` and show the contents of the directory asked by the user in a cascading format as follows. Suppose we `g++ -o dir.out dir.cpp`, the results of the following command are shown as follows.

   ```
   ./dir.out cli2/IT279/

   /home/cli2/IT279/
   IT279
        Asg1
             a.out
             Asg1.cpp
             myArray.cpp
             myArray.h
             testAsg1.cpp
             testasg1.sh
        Asg2
             allUsersDir.txt
             dir.out
             dir.cpp
             dirTree.cpp
             Tree.h
        copy279all.sh
   ```

If the user does not provide a directory as an argument to `dir.out`, then the entire `/home/` directory will be shown on the screen. If the asked directory does not exist, you are free to handle this situation in any way, but your program should not crash.

5. Standard output to submit will be the results of the following two commands:

```
./dir.out yourULID/IT279/

./dir.out yourULID/public/
```

Replace `yourULID` with your Unix account's user id.

Secret directory under `public`

After you've finished your work, or have decided that what you have is the final version for me to grade, you have to copy all your programs to a secret directory under your `public/IT279myWork` so I can grade them (i.e., compile, run, and check the codes). Select a secret name, say "`peekapoo`" as an example (you should chose your own), and that will be the secret directory.

You can simply copy my `/home/cli2/public/IT279/copy279all.sh` and use the following command line to run it. (You can put `copy279all.sh` in any directory.)

```
bash copy279all.sh peekapoo
```

This is important. Follow the instruction in 1. to prepare your work. All your works for this assignment should be put under the required directory, otherwise `copy279all.sh` can't find your works.

You can update your `allUsersDir.txt` and use your program to check what are in your `public`.

Final Words:

You have to follow the submission guidelines, i.e., cover page (that contains assignment number, student's names, student **ULID**, and secret directory), summary, source code(optional), output, folder, and so on. **No late work will be accepted.**