COMP 311: Project #3

Due on January 18, 2021 at 11:59 PM (Before Midnight) on ITC

Under your home directory, create a directory called `project3_[university_id]` and inside the directory, write a *shell script* called `mysh`. When `mysh` is executed, it displays the prompt: `myprompt >` and accepts one of the following six commands:

1. `myfinger` *name1 name2 name3 ...*

   - This command will display the *login name*, *full name*, *home directory*, and the *default shell* for each of the names given on the command line.
   - If one of the names is **not found**, it should display an *error message* corresponding to that name.
   - If `myfinger` is entered **without** any arguments, then the *login name*, *full name*, *home directory*, and the *default shell* should be displayed for all the users **logged in** to the system at that time. You can use the `who` command to retrieve all users logged in to the system.
   - You are **not allowed** to use the `finger` command in implementing this command.
   - You can rely on the format set by `prj2_passwd` file from Project 2 to implement this command.
   - Example usage:

   ```
   myprompt > myfinger u1150129
   Login Name: u1150129
   Full Name:  Mohammad Kahala
   Home Directory: /home/students/comp311/u1150129
   Default Shell: /bin/bash
   myprompt >
   ```

2. `mywhich` *command1 command2 command3 ...*

   - This command will check each directory listed in user's PATH variable for each of the commands given as arguments, then display the **full path of each command** (file) found in the PATH.
   - If a given command (file) cannot be found in any of the directories listed in PATH, then it should display a message saying: `no such command`.
   - This command takes at least one argument, so in case of no arguments, you should display a proper error message.
   - You are **not allowed** to use the `which` command in implementing this command.
   - Example usage:

   ```
   local-machine$ echo $PATH
   /usr/bin:/bin:/usr/sbin
   local-machine$ ./mysh
   myprompt > mywhich command1
   /usr/bin/command1
   /usr/sbin/command1
   myprompt >
   ```

3. `mycp` *srcfile dstfile*

   - This command only copies a given source file name (if it exists) to a given destination file name.
   - If no file names are given it should display an error message. You also have to handle the case where the given arguments corresponds to directories or non-existence files.
   - You are **not allowed** to use the `cp` command in implementing this command. Think of another way to copy the contents of a file to another file.

- Example usage:

```
local-machine$ ls
work ffff aaaa bbbb
local-machine$ ./mysh
myprompt > mycp file1 file2
Error: file1 does not exit.
myprompt > mycp aaaa cccc
Success: File [aaaa] copied to file [cccc].
myprompt >
```

4. `mydel` *filename(s) directoryname(s)*

- This command deletes any file(s) or directorie(s) given as arguments (if they exist).
- The command takes at least one argument. In case of no arguments, proper error message should be displayed.
- Example usage:

```
local-machine$ ls
work ffff aaaa bbbb
local-machine$ ./mysh
myprompt > mydel work ffff wrong
[work] directory is deleted.
[ffff] file is deleted.
[wrong] file does not exit.
myprompt >
```

5. `myhelp` *command1 command2 command3 ...*

- This command displays one line of help/usage for any of the six commands (`myfinger`, `mywhich`, `mycp`, `mydel`, `myhelp`, and `myexit`) which can be passed as arguments. You are free to choose proper wording for each help line.
- The command may take more than one argument.
- If no arguments are entered, the command should display help for all six commands: `myfinger`, `mywhich`, `mycp`, `mydel`, `myhelp`, and `myexit`
- Example usage:

```
myprompt > myhelp mydel myexit
[mydel] This command deletes any existence file(s)/directory(s) given as arguments.
[myexit] This command lets the users exit 'mysh' script and return to original shell.
myprompt >
```

6. `myexit`

- This command lets the users exit the shell script `mysh` and return to their original shell.
- This command takes no arguments.
- Example usage:

```
myprompt > myexit
local-machine$
```

7. If any commands other than the six commands above are entered, mysh should display the message: Command not found. An example usage:

   ```
   myprompt > hello
   Command not found
   myprompt >
   ```

Be sure to use the designated names for your shell script files as well as the commands. Make sure to follow proper modular design and use proper indentation to make your script readable. Also, be sure to include any necessary error checking. The script will keep prompting the user to enter any of the six commands until the command myexit is entered.

Submission

To successfully submit your solution to this project and be considered for grading, you **MUST** upload **two artifacts** to **Comp311 meta section** under title "**Final Project Turned in here**" on ITC before the deadline:

1. A script log named project3_[university_id].log containing the steps you followed to do the required tasks. Similar to what you do in regular lab tasks.
2. A compressed file named project3_[university_id].tar that contains the script and any other test files you have created. To create a compressed file:

   Go to the **home directory** and type the command:

   ```
   tar -cvf project3_[university_id].tar project3_[university_id]
   ```

Important Notes

1. You should do all the work above completely on your own. Working with anybody else on any part of this project will result in a zero grade.
2. No projects will be accepted after the specified deadline for any reason.
3. Projects that do not have the corresponding log file turned in will NOT be graded.