Test Case #1

Sample Input File

testcase1.txt
| ```
useradd root ya84*_o
login root ya84*_o
useradd alice Wvu_4_Life
useradd bob SHHHsecret
groupadd students
usergrp alice students
usergrp bob students
useradd tom geek_247
logout
login alice Wvu_4_Life
mkfile file1.txt
write file1.txt Text from Alice in file1
mkfile file2.txt
write file2.txt Text from Alice in file2
chmod file2.txt rw- rw- r--
chgrp file2.txt students
logout
login tom geek_247
read file1.txt
read file2.txt
write file2.txt Text from Tom in file2
login bob SHHsecret
logout
login bob SHHsecret
login bob SHHHsecret
write file2.txt Text from Bob in file2
read file2.txt
logout
login root ya84*_o
chown file1.txt tom
ls file1.txt
logout
login tom geek_247
write file1.txt Text from Tom in file1
read file1.txt
logout
end
``` |

Line-by-Line Expected Program Execution

```
>useradd root ya84*_o
```
- Creates privileged user `root` by adding entry to *accounts.txt file* containing username and plaintext (or hashed) password
- Display message to terminal: `"User root created"`
- Log message to *audit.txt* file: `"User root created"`

```
>login root ya84*_o
```
- Locates entry for user `root` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `root` into system
- Display message to terminal: `"User root logged in"`
- Log message to *audit.txt* file: `"User root logged in"`

```
>useradd alice Wvu_4_Life
```
- Creates user `alice` by adding entry to *accounts.txt* file containing username and plaintext (or hashed) password
- Display message to terminal: `"User alice created"`
- Log message to *audit.txt* file: `"User alice created"`

```
>useradd bob SHHHsecret
```
- Creates user `bob` by adding entry to *accounts.txt file* containing username and plaintext (or hashed) password
- Display message to terminal: `"User bob created"`
- Log message to *audit.txt* file: `"User bob created"`

```
>groupadd students
```
- Creates group `students` without any users
- Display message to terminal: `"Group students created"`
- Log message to *audit.txt* file: `"Group students created"`

```
>usergrp alice students
```
- Adds user `alice` to group `students`
- Display message to terminal: `"User alice added to group students"`
- Log message to *audit.txt* file: `"User alice added to group students"`

```
>usergrp bob students
```
- Adds user `bob` to group `students`
- Display message to terminal: `"User bob added to group students"`
- Log message to *audit.txt* file: `"User bob added to group students"`

> `useradd tom geek_247`
- Creates user `tom` by adding entry to *accounts.txt file* containing username and plaintext (or hashed) password
- Display message to terminal: `"User tom created"`
- Log message to *audit.txt* file: `"User tom created"`

> `logout`
- Logs out user `root`
- Display message to terminal: `"User root logged out"`
- Log message to *audit.txt* file: `"User root logged out"`

> `login alice Wvu_4_Life`
- Locates entry for user `alice` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `alice` into system
- Display message to terminal: `"User alice logged in"`
- Log message to *audit.txt* file: `"User alice logged in"`

> `mkfile file1.txt`
- Creates empty file named *file1.txt* with default permissions and `alice` as the owner
- Display message to terminal: `"File file1.txt with owner alice and default permissions created"`
- Log message to *audit.txt* file: `"File file1.txt with owner alice and default permissions created"`

> `write file1.txt Text from Alice in file1`
- Appends supplied text as new line to *file1.txt*
- Display message to terminal: `"User alice wrote to file1.txt: Text from Alice in file1"`
- Log message to *audit.txt* file: `"User alice wrote to file1.txt: Text from Alice in file1"`

> `mkfile file2.txt`
- Creates empty file named *file2.txt* with default permissions and `alice` as the owner
- Display message to terminal: `"File file2.txt with owner alice and default permissions created"`
- Log message to *audit.txt* file: `"File file2.txt with owner alice and default permissions created"`

>`write file2.txt Text from Alice in file2`
- Appends supplied text as new line to *file2.txt*
- Display message to terminal: "`User alice wrote to file2.txt: Text from Alice in file2`"
- Log message to *audit.txt* file: "`User alice wrote to file2.txt: Text from Alice in file2`"

>`chmod file2.txt rw- rw- r--`
- Sets file permissions for *file2.txt* to allow read and write for owner and group, but only read for public
- Display message to terminal: "`Permissions for file2.txt set to rw- rw- r-- by alice`"
- Log message to *audit.txt* file: "`Permissions for file2.txt set to rw- rw- r-- by alice`"

>`chgrp file2.txt students`
- Sets group for *file2.txt* to `students`
- Display message to terminal: "`Group for file2.txt set to students by alice`"
- Log message to *audit.txt* file: "`Group for file2.txt set to students by alice`"

>`logout`
- Logs out user `alice`
- Display message to terminal: "`User alice logged out`"
- Log message to *audit.txt* file: "`User alice logged out`"

>`login tom geek_247`
- Locates entry for user `tom` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `tom` into system
- Display message to terminal: "`User tom logged in`"
- Log message to *audit.txt* file: "`User tom logged in`"

>`read file1.txt`
- Display message to terminal: "`User tom denied read access to file1.txt`"
- Log message to *audit.txt* file: "`User tom denied read access to file1.txt`"

>`read file2.txt`
- User `tom` successfully reads *file2.txt*
- Display message to terminal:
  "`User tom reads file2.txt as: Text from Alice in file2`"

- Log message to *audit.txt* file:
  ```
        "User tom reads file2.txt as:
         Text from Alice in file2"
  ```

> `write file2.txt Text from Tom in file2`
- Display message to terminal: "`User tom denied write access to file2.txt`"
- Log message to *audit.txt* file: "`User tom denied write access to file2.txt`"

> `login bob SHHsecret`
- Display message to terminal: "`Login failed: simultaneous login not permitted`"
- Log message to *audit.txt* file: "`Login failed: simultaneous login not permitted`"

> `logout`
- Logs out user `tom`
- Display message to terminal: "`User tom logged out`"
- Log message to *audit.txt* file: "`User tom logged out`"

> `login bob SHHsecret`
- Display message to terminal: "`Login failed: invalid username or password`"
- Log message to *audit.txt* file: "` Login failed: invalid username or password `"

> `login bob SHHHsecret`
- Locates entry for user `bob` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `bob` into system
- Display message to terminal: "`User bob logged in`"
- Log message to *audit.txt* file: "`User bob logged in`"

\> `write file2.txt Text from Bob in file2`
- Appends supplied text as new line to *file2.txt*
- Display message to terminal: "`User bob wrote to file2.txt: Text from Bob in file2`"
- Log message to *audit.txt* file: "`User bob wrote to file2.txt: Text from Bob in file2`"

\> `read file2.txt`
- User `bob` successfully reads *file2.txt*
- Display message to terminal:
      "`User bob reads file2.txt as:`
       `Text from Alice in file2`
       `Text from Bob in file2`"
- Log message to *audit.txt* file:
      "`User bob reads file2.txt as:`
       `Text from Alice in file2`
       `Text from Bob in file2`"

\> `logout`
- Logs out user `bob`
- Display message to terminal: "`User bob logged out`"
- Log message to *audit.txt* file: "`User bob logged out`"

\> `login root ya84*_o`
- Locates entry for user `root` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `root` into system
- Display message to terminal: "`User root logged in`"
- Log message to *audit.txt* file: "`User root logged in`"

\> `chown file1.txt tom`
- Changes owner of *file.txt* to user `tom`
- Display message to terminal: "`Owner of file1.txt changed to tom`"
- Log message to *audit.txt* file: "`Owner of file1.txt changed to tom`"

\> `ls file1.txt`
- Displays the owner, group, and permissions for *file1.txt*
- Display message to terminal: "`file1.txt: tom nil rw- --- ---`"
- Log message to *audit.txt* file: "`file1.txt: tom nil rw- --- ---`"

\> `logout`
- Logs out user `root`
- Display message to terminal: "`User root logged out`"
- Log message to *audit.txt* file: "`User root logged out`"

> `login tom geek_247`
- Locates entry for user `tom` in *accounts.txt* file and verifies that supplied password (or hash of supplied password) matches the one in the file
- Successfully logs user `tom` into system
- Display message to terminal: `"User tom logged in"`
- Log message to *audit.txt* file: `"User tom logged in"`

> `write file1.txt Text from Tom in file1`
- Appends supplied text as new line to *file1.txt*
- Display message to terminal: `"User tom wrote to file1.txt: Text from Tom in file1"`
- Log message to *audit.txt* file: `"User tom wrote to file1.txt: Text from Tom in file1"`

> `read file1.txt`
- User `tom` successfully reads *file1.txt*
- Display message to terminal:
  ```
          "User tom reads file1.txt as:
          Text from Alice in file1
          Text from Tom in file1"
  ```
- Log message to *audit.txt* file:
  ```
          "User tom reads file1.txt as:
          Text from Alice in file1
          Text from Tom in file1"
  ```

> `logout`
- Logs out user `tom`
- Display message to terminal: `"User tom logged out"`
- Log message to *audit.txt* file: `"User tom logged out"`

> `end`
- Program writes out the *groups.txt* and *files.txt* files, performs necessary clean-up, and terminates

Final Content of User Created Files

```
file1.txt
Text from Alice in file1
Text from Tom in file1
```

```
file2.txt
Text from Alice in file2
Text from Bob in file2
```

groups.txt Content

```
groups.txt
students: alice bob
```

files.txt Content

```
files.txt
file1.txt: tom nil rw- --- ---
file2.txt: alice students rw- rw- r--
```

accounts.txt Content *(An example format with plain ASCII passwords)*

```
accounts.txt
root ya84*_o
alice Wvu_4_Life
bob SHHHsecret
tom geek_247
```

Audit Log Content

```
audit.txt
```

```
User root created
User root logged in
User alice created
User bob created
Group students created
User alice added to group students
User bob added to group students
User tom created
User root logged out
User alice logged in
File file1.txt with owner alice and default permissions created
User alice wrote to file1.txt: Text from Alice in file1
File file2.txt with owner alice and default permissions created
User alice wrote to file2.txt: Text from Alice in file2
Permissions for file2.txt set to rw- rw- r-- by alice
Group for file2.txt set to students by alice
User alice logged out
User tom logged in
User tom denied read access to file1.txt
User tom reads file2.txt as:
Text from Alice in file2
User tom denied write access to file2.txt
Login failed: simultaneous login not permitted
User tom logged out
Login failed: invalid username or password
User bob logged in
User bob wrote to file2.txt: Text from Bob in file2
User bob reads file2.txt as:
Text from Alice in file2
Text from Bob in file2
User bob logged out
User root logged in
Owner of file1.txt changed to tom
file1.txt: tom nil rw- --- ---
User root logged out
User tom logged in
User tom wrote to file1.txt: Text from Tom in file1
User tom reads file1.txt as:
Text from Alice in file1
Text from Tom in file1
User tom logged out
```