

# BÀI KIỂM TRA LẬP TRÌNH SHELL LINUX

## Bài 1: Viết chương trình Shell hỗ trợ việc tạo nhiều user cùng một lúc trên HĐH Linux

- Tên chương trình Shell Script: **AddMultUsers.sh**
- Input chương trình Shell Script có cú pháp: Mỗi dòng là Username và Password của User cần thêm vào HĐH. (Username và Password cách nhau bởi khoảng trắng)

Ví dụ:

Teo 123456  
Ti abcdef  
CuBin 123456?@

- Output chương trình Shell Script: thông báo tạo thành công hay không.
- Cơ chế thực hiện chương trình:
  - + Đọc danh sách Username và Password từ file input.
  - + Kiểm tra Username[i] thêm vào có tồn tại trong HĐH chưa?
    - \* Nếu chưa tồn tại, thì thêm User mới vào.
    - \* Nếu tồn tại rồi, xuất thông báo Username[i] đã tồn tại.
    - \* *Ghi chú: Nếu Username[i] tồn tại trong HĐH mà Username[i+1] chưa tồn tại thì vẫn thêm Username[i+1] vào và thông báo Username[i] đã tồn tại.*

Ví dụ: Kết quả thực hiện chương trình:

Đã thêm user Teo  
User Ti đã tồn tại  
Đã thêm user CuBin

- Cú pháp chạy chương trình Shell: **./AddMultUsers.sh <file chứa danh sách User>**

Ví dụ: ./AddMultUser.sh input\_user.txt

## Bài 2: Viết chương trình Shell hỗ trợ việc thêm các User vào nhóm người dùng mong muốn.

- Tên chương trình Shell Script: **AddUserToGroup.sh**
- Input chương trình có cú pháp: Mỗi dòng là Username của User cần được thêm vào Group (group phụ).

Ví dụ:

Ti  
Teo  
CuBin  
BeBo

- Output chương trình Shell Script: thông báo tạo thành công hay không.
- Cơ chế thực hiện chương trình:

- + Đọc danh sách Username từ file input.

- + Kiểm tra Username[i] có tồn tại trong Group chưa?

- \* Nếu chưa tồn tại thì thêm User đó vào Group

- \* Nếu tồn tại rồi, thông báo User đó đã thuộc Group rồi.

- \* Ghi chú: Nếu Username[i] tồn tại trong Group, mà Username[i+1] chưa tồn tại thì vẫn thêm Username[i+1] vào trong Group và thông báo Username[i] đã tồn tại rồi.*

Ví dụ: Kết quả thực thi chương trình:

Đã thêm user Ti vào Group  
User Teo đã có trong Group  
Đã thêm user CuBin vào Group  
Đã thêm user BeBo vào Group

- Cú pháp chạy chương trình Shell:

**./AddUserToGroup.sh <file chứa danh sách User> <tên Group>**

Ví dụ: ./AddUserToGroup.sh input\_user.txt Ky\_Thuat

### Bài 3: Viết chương trình Shell hỗ trợ phân quyền lên nhiều file cùng một lúc. (không có thay đổi quyền sở hữu)

- Tên chương trình Shell Script: **ChangeModFiles.sh**

- Input chương trình Shell Script có cú pháp: Mỗi dòng là tên tập tin và quyền truy xuất trên tập tin. (Quyền truy xuất tập tin được thể hiện bằng số)

Ví dụ:	Ghi chú
/home/sinhvien/thongbao.txt 753	Đường dẫn đầy đủ của tập tin
Desktop/bantin.doc 550	Thư mục con Desktop của thư mục hiện hành có chứa tập tin bantin.doc
Thongtin.pdf 555	Thư mục hiện hành có chứa tập tin Thongtin.pdf

- Output chương trình Shell Script: liệt kê lại quyền hạn của tập tin đã thay đổi (giống lệnh **ls -l**)

- Cơ chế thực hiện chương trình:

+ Đọc danh sách tập tin và quyền từ file input.

+ Kiểm tra tập tin có tồn tại hay không?

\* Nếu tồn tại, thì phân quyền cho tập tin.

\* Nếu chưa tồn tại, thông báo cho rằng tập tin chưa tồn tại.

\* *Ghi chú: Nếu Tập tin thứ [i] không tồn tại mà Tập tin thứ [i+1] tồn tại thì vẫn phân quyền cho Tập tin thứ [i+1] vào và thông báo Tập tin thứ [i] không tồn tại.*

Ví dụ input:	Ví dụ output:
/home/sinhvien/thongbao.txt 753	<pre>-rwxr-x-wx. 1 root root 0 Apr 28 10:21 /home/sinhvien/thongbao.txt</pre>
Thongtin.pdf 555	Tập tin Thongtin.pdf không tồn tại
Desktop/bantin.doc 550	<pre>-r-xr-x---. 1 root root 0 Apr 28 10:23 Desktop/bantin.doc</pre>

- Cú pháp chạy chương trình Shell: **./ChangeModFiles.sh <danh sách các Files>**

Ví dụ: **./ChangeModFiles.sh input\_file.txt**