



BÁO CÁO

MÔN HỌC: HỆ ĐIỀU HÀNH

ĐỒ ÁN 1: SIMPLE SHELL

I. Danh sách thành viên

STT	MSSV	Họ và tên
1	1612866	Vũ Duy Khánh
2	1712457	Nguyễn Khánh Hoàng
3	1712460	Phạm Minh Hoàng

II. Mô tả tổ chức, thiết kế đồ án:

Đồ án “chương trình Shell đơn giản” (Simple Shell) mô phỏng hoạt động của một command-line interface (CLI) - giao diện dòng lệnh, trong đó người dùng có thể nhập lệnh vào và thực thi lệnh. Chương trình Simple Shell hỗ trợ một số lệnh cơ bản của hệ điều hành Ubuntu Linux như `cd`, `cat`, `less`, `ls`, `pwd`, `touch`, `mkdir`, `rm`, `find`, ... Chương trình sẽ hỗ trợ *input*, *output redirection* cũng như là *pipes* - một dạng giao tiếp (trao đổi, chuyển tiếp dữ liệu, output của tiến trình này có thể trở thành input của tiến trình tiếp theo nó) giữa các câu lệnh / tiến trình với nhau.

Chương trình được thiết kế hoạt động theo cơ chế như sau: Thực hiện một vòng lặp vô hạn trong đó người dùng được thấy một dấu nhắc nhập lệnh trên giao diện của chương trình. Dấu nhắc lệnh : ***osh>*** ... Tại đây người dùng sẽ nhập lệnh vào và nhấn Enter. Ví dụ : ***osh> cat prog.c*** (Lệnh này hiển thị nội dung trong file `prog.c` lên màn hình Terminal/Shell). Xảy ra các trường hợp sau :

- Nếu người dùng gõ “exit” thì chương trình kết thúc.lệnh
- Nếu người dùng không gõ gì mà chỉ nhấn Enter thì tiếp tục nhắc lệnh.
- Nếu người dùng gõ “!!” : Nếu trong lịch sử chưa có lệnh nào thì thông báo “No commands in history!”, ngược lại in ra màn hình lệnh gần đây nhất và thực thi nó.
- Để thi hành một giao diện shell (dòng lệnh), đầu tiên cho tiến trình cha đọc vào những gì mà người dùng nhập từ dòng lệnh và sau đó tạo ra một tiến trình con riêng biệt rồi để nó thực hiện lệnh này. Trừ khi có điều kiện gì khác được chỉ định, tiến trình cha sẽ chờ cho tiến trình con hoàn thành

xong xuôi rồi mới tiếp tục chạy. Tuy nhiên, các UNIX shell điển hình

cũng cho phép các tiến trình con chạy ở dưới nền, tức là chạy đồng thời. Để đạt được điều này, chúng ta thêm một dấu & ở phía cuối của dòng lệnh. Do vậy, nếu chúng ta viết câu lệnh ở phía trên thành như sau : **osh> cat prog.c &** thì tiến trình cha và con sẽ chạy đồng thời với nhau. Tiến trình con riêng biệt sẽ được tạo bằng cách dùng lệnh hệ thống fork(), và lệnh của người dùng sẽ được thực thi bằng cách sử dụng một trong những lệnh hệ thống trong nhóm lệnh exec() dành cho nền tảng Linux.

Đồ án được tổ chức thành 4 phần :

1. Thực thi lệnh được nhập vào bởi người dùng : Chương trình sẽ tạo một tiến trình con tách biệt với tiến trình cha (chức năng hiển thị dấu nhắc lệnh) và cho tiến trình con này thực thi lệnh được nhập vào. Nếu tham số (argument) cuối cùng của câu lệnh là ký tự "&" thì chương trình sẽ cho phép tiến trình con chạy đồng thời với tiến trình cha. Ngược lại, nếu không có "&", tiến trình cha phải đợi tiến trình con kết thúc xong mới tiếp tục chạy.
2. Tính năng **History** (Lịch sử) : Chương trình sẽ lưu lại câu lệnh gần nhất mà người dùng nhập vào, nếu người dùng muốn thực hiện lại lệnh đó thì chỉ cần gõ **!!** . Trường hợp chưa có lệnh nào thì thông báo ra màn hình.
3. Tính năng **Input / Output Redirection** : Chương trình hỗ trợ người dùng nhập vào lệnh với dạng như : **ls > out.txt** hay **sort < array.txt** .
 - ▶ Tính năng Output Redirection : với lệnh **ls > out.txt**, kết quả thực hiện lệnh ls sẽ được xuất ra file out.txt thay vì hiện lên màn hình Shell.
 - ▶ Tính năng Input Redirection : với lệnh **sort < array.txt**, nội dung của file array.txt sẽ được truyền vào làm tham số (argument) của lệnh **sort**, lệnh **sort** này được thực thi và kết quả sẽ được hiện lên màn hình Shell.
4. Tính năng liên lạc giữa các tiến trình con với nhau - **pipe** : Chương trình hỗ trợ người dùng nhập vào lệnh dưới dạng : **osh> ls -l | less** . Output của

lệnh **ls -l** thay vì được hiển thị ra màn hình shell thì sẽ trở thành input và được truyền vào lệnh **less**. Khi đó lệnh **less** sẽ hiển thị nội dung văn bản của kết quả mà câu lệnh **ls -l** đã thực hiện.

Phân công công việc cho các thành viên trong nhóm :

MSSV	Họ và tên	Phân công nhiệm vụ	Mức độ hoàn thành
1612866	Vũ Duy Khánh	Cài đặt chức năng Pipe	100%
1712457	Nguyễn Khánh Hoàng	- Xử lý tách chuỗi nhập vào và thực thi câu lệnh - Xử lý thực thi lệnh <i>cd</i> - Cài đặt tính năng Input/Output Redirection	100%
1712460	Phạm Minh Hoàng	- Cài đặt tính năng lịch sử lệnh gần nhất - Xử lý việc chạy các tiến trình con đồng thời với tiến trình cha	100%

III. Kiểm thử các chức năng với một số câu lệnh:

- Đầu tiên, ta test thử chức năng history “!!” khi chưa thực hiện câu lệnh nào:

```
osh>!!
No commands in history!
osh>
```

- Ta thử lệnh **ls -l**:

```
osh>ls -l
total 36
drwxrwxr-x 2 ubuntu ubuntu 4096 Th10  6 17:51 aaa
-rw-rw-r-- 1 ubuntu ubuntu  12 Th10  6 18:10 aaa.txt
-rwxrwxr-x 1 ubuntu ubuntu 14232 Th10  6 16:19 project
-rw-rw-r-- 1 ubuntu ubuntu  5238 Th10  6 16:19 project.c
drwxrwxr-x 6 ubuntu ubuntu 4096 Th10  6 16:56 Test
```

- Test lại chức năng history “!!”:

```
osh>!!
osh>ls -l
total 36
drwxrwxr-x 2 ubuntu ubuntu 4096 Th10  6 17:51 aaa
-rw-rw-r-- 1 ubuntu ubuntu  12 Th10  6 18:10 aaa.txt
-rwxrwxr-x 1 ubuntu ubuntu 14232 Th10  6 16:19 project
-rw-rw-r-- 1 ubuntu ubuntu 5238 Th10  6 16:19 project.c
drwxrwxr-x 6 ubuntu ubuntu 4096 Th10  6 16:56 Test
```

- Ta thử lệnh cat:

+ cat hello.c:

```
osh>cat hello.c
#include <stdio.h>

int main()
{
    printf("Hello world!\n");
    return 0;
}
```

- Ta thử một số lệnh khác như pwd, mkdir, rmdir, touch, rm

+ pwd:

```
osh>pwd
/home/ubuntu/Desktop
```

+ mkdir (tạo directory thành công):

```
osh>mkdir abc
osh>
```



+ rmdir (xóa directory thành công):

```
osh>rmdir abc
```

+ touch (tạo thành công):

```
osh>touch abc.txt
```



+ rm (xóa thành công):

```
osh>rm abc.txt
```

+ chmod -help:

```
osh>chmod --help
Usage: chmod [OPTION]... MODE[,MODE]... FILE...
       or: chmod [OPTION]... OCTAL-MODE FILE...
       or: chmod [OPTION]... --reference=RFIL FILE...
Change the mode of each FILE to MODE.
With --reference, change the mode of each FILE to that of RFIL.

  -c, --changes          like verbose but report only when a change is made
  -f, --silent, --quiet  suppress most error messages
  -v, --verbose          output a diagnostic for every file processed
                        --no-preserve-root  do not treat '/' specially (the default)
                        --preserve-root    fail to operate recursively on '/'
                        --reference=RFIL   use RFIL's mode instead of MODE values
  -R, --recursive        change files and directories recursively
  --help                display this help and exit
  --version              output version information and exit

Each MODE is of the form '[ugoa]*([-+=]([rwxXst]*|[ugo]))+|[-+=][0-7]+'

GNU coreutils online help: <http://www.gnu.org/software/coreutils/>
Full documentation at: <http://www.gnu.org/software/coreutils/chmod>
or available locally via: info '(coreutils) chmod invocation'
```

- Một số lệnh như find, gzip, diff, df:

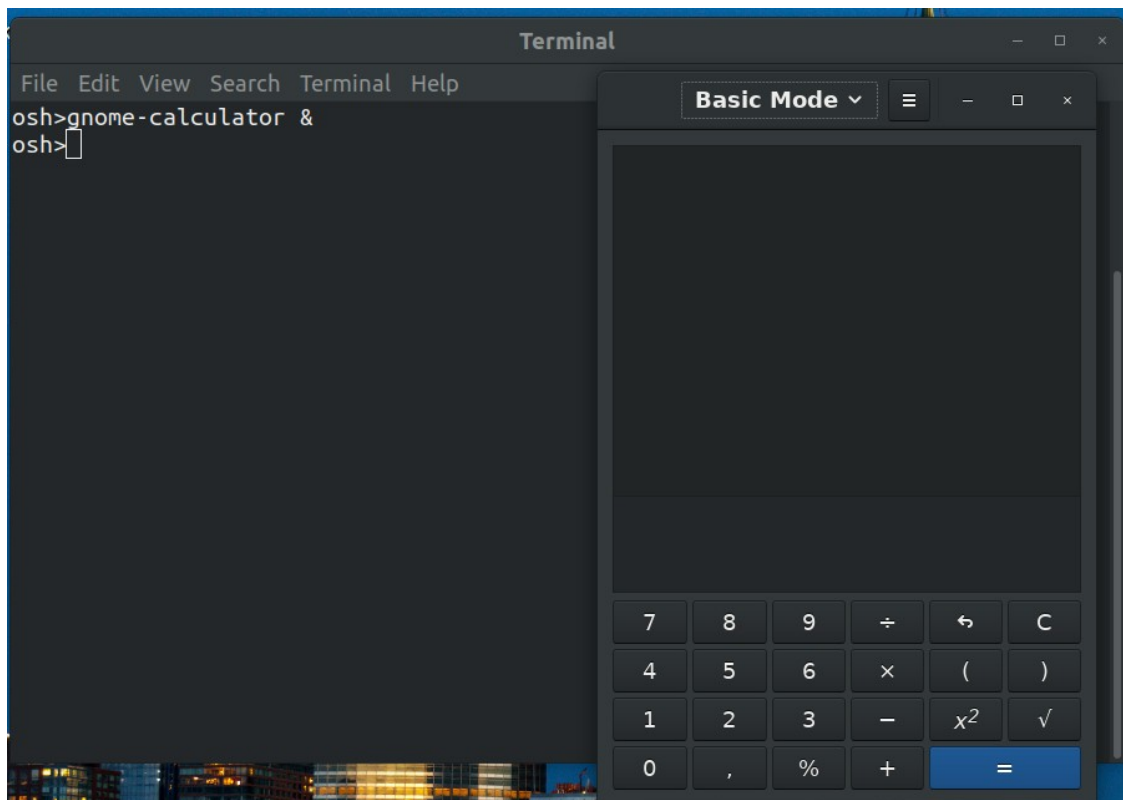
+ find:

```
osh>find aaa.txt
aaa.txt
```

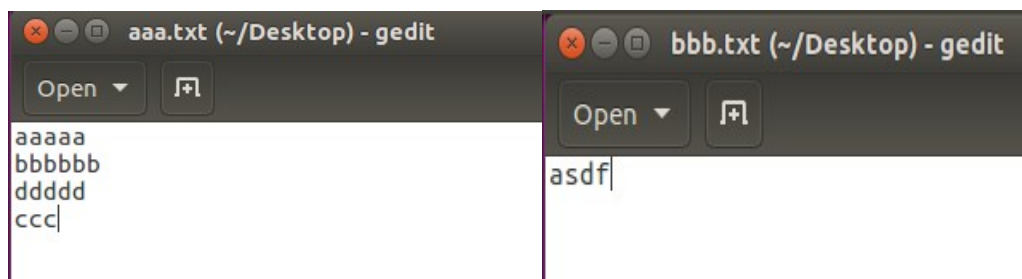
+ gzip:


```
osh>gzip -k aaa.txt
gzip: aaa.txt: No such file or directory
osh>gzip -k bbb.txt
osh>ls
bbb.txt      main      output.txt  project     Test
bbb.txt.gz   main.c    out.txt     project.c
```

- Chạy đồng thời tiến trình cha (shell) và tiến trình con (gnome-calculator) :



+ diff



```
osh>diff -y -W 70 aaa.txt bbb.txt --suppress-common-lines
aaaaa      |      asdf
bbbbbb     <
ddddd      <
ccc        <
```

+ df:

```
osh>df -h -x squashfs
Filesystem      Size  Used Avail Use% Mounted on
udev            1,9G   0    1,9G   0% /dev
tmpfs           384M  6,2M  378M   2% /run
/dev/sda4       18G   5,8G   12G   34% /
tmpfs           1,9G  245M   1,7G  13% /dev/shm
tmpfs           5,0M   4,0K   5,0M   1% /run/lock
tmpfs           1,9G   0    1,9G   0% /sys/fs/cgroup
tmpfs           384M   80K   384M   1% /run/user/1000
```

+ exit:

```
osh>exit
ubuntu@ubuntu-Inspiron-3537:~/Desktop$
```

- Tiếp tục test Input, Output Redirection:

+ ls > out.txt

```
osh>ls > out.txt
```



```
File Edit View Search
osh>ls > out.txt
osh>cat out.txt
array.txt
func
funcs.c
funcs.h
out.txt
shell
shell.c
osh>
```

+ sort:


```
aaa.txt (~/Desktop) - gedit
Open [v] [f+l]
aaa.txt
aaaaa
bbbbbb
dddddd
ccc
```

```
osh>sort < aaa.txt
aaaaa
bbbbbb
ccc
dddddd
```

+ cd :

```
Terminal
File Edit View Search Terminal Help
osh>gnome-calculator &
osh>pwd
/home/hoang1807/Documents/source-code/SimpleShell
osh>cd ~
Current directory : /home/hoang1807
osh>cd Desktop
Current directory : /home/hoang1807/Desktop
osh>
```

- Test Pipe: (với lệnh `ls -l | less`)

```
total 68
-rw-rw-r-- 1 ubuntu ubuntu 5 Th10 9 23:42 bbb.txt
-rw-rw-r-- 1 ubuntu ubuntu 33 Th10 9 23:42 bbb.txt.gz
-rwxrwxr-x 1 ubuntu ubuntu 14824 Th10 10 00:50 main
-rw-rw-r-- 1 ubuntu ubuntu 10211 Th10 10 00:44 main.c
----- 1 ubuntu ubuntu 0 Th10 6 19:53 output.txt
----- 1 ubuntu ubuntu 43 Th10 6 19:12 out.txt
-rwxrwxr-x 1 ubuntu ubuntu 14232 Th10 6 16:19 project
-rw-rw-r-- 1 ubuntu ubuntu 5238 Th10 6 16:19 project.c
drwxrwxr-x 6 ubuntu ubuntu 4096 Th10 6 16:56 Test
(END)
```