

**HỆ ĐIỀU HÀNH**

**Đồ án 1**

# **SIMPLE SHELL**

## **Nhóm 2**

1712542 – Hoàng Anh Kiệt

1712525 - Nguyễn Ngọc Minh Khánh

1712531 – Hồ Anh Khoa



Khoa Công nghệ Thông tin  
Đại học Khoa học Tự nhiên TP HCM  
Tháng 10/2019

**HỆ ĐIỀU HÀNH**

**Đồ án 1**

# **SIMPLE SHELL**

## **Nhóm 2**

1712542 – Hoàng Anh Kiệt

1712525 - Nguyễn Ngọc Minh Khánh

1712531 – Hồ Anh Khoa

**GVHD: Trần Trung Dũng**



Khoa Công nghệ Thông tin  
Đại học Khoa học Tự nhiên TP HCM  
Tháng 10/2019

# Mục lục

I.	Phần đã làm .....	4
II.	Các hàm chính trong chương trình .....	4
III.	Hướng dẫn cách chạy chương trình.....	5
IV.	Minh họa các yêu cầu trong đồ án.....	7
V.	Tham khảo .....	10

## I. Phần đã làm

- Tạo tiến trình con và thực hiện các câu lệnh trong tiến trình con. Thực thi cả trong trường hợp có dấu '&'.
- Cung cấp chức năng lịch sử khi người dùng nhấn '!!'.
- Chuyển hướng dữ liệu.
- Cho phép tiến trình cha có thể giao tiếp với tiến trình con qua một đường ống.

## II. Các hàm chính trong chương trình

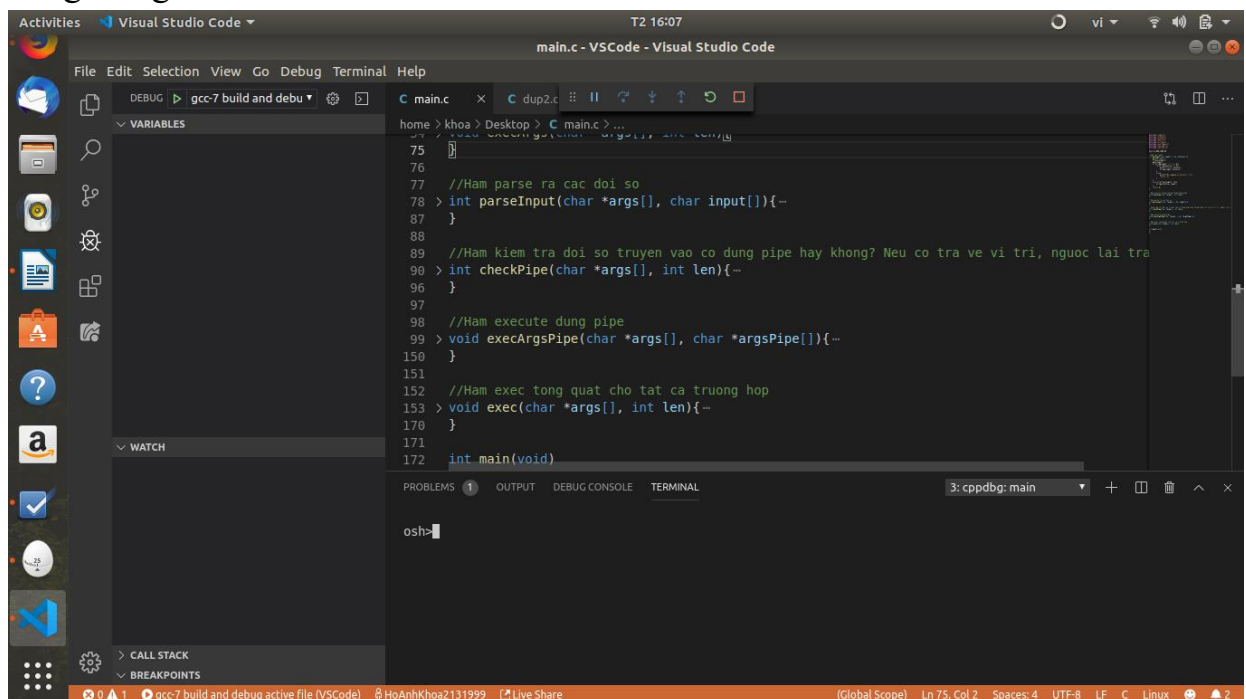
Tên hàm	Chức năng	Mô tả
<code>int getInput(char input[], char history[])</code>	<ul style="list-style-type: none"><li>• Nhận lệnh từ người dùng bàn phím.</li><li>• Chỉnh sửa lịch sử.</li></ul>	<ul style="list-style-type: none"><li>• Dùng hàm <b>gets</b> để đọc những gì người dùng nhập. Chuyển vào chuỗi <b>input</b></li><li>• Kiểm tra xem nếu là '!!' thì gán <b>input</b> vào bằng <b>history</b>.</li></ul>
<code>int parseInput(char *args[], char input[])</code>	Biến chuỗi <b>input</b> thành mảng các token <b>args</b> (phân cách nhau bởi dấu cách).	Sử dụng hàm <b>strtok()</b> phân cách bằng dấu cách (' ').
<code>int checkPipe(char *args[], int len)</code>	Kiểm tra xem có dấu ' ' hay không?.	Duyệt qua toàn bộ các token trong mảng nếu có phần tử nào bằng " " thì trả về vị trí đó
<code>void execArgs(char *args[], int len)</code>	Thực thi một lệnh bằng cách thông thường. Không có chức năng <b>pipe</b> .  Ví dụ: <b>osh&gt; ls -l</b>	<ul style="list-style-type: none"><li>• Dùng hàm <b>fork()</b> để chia tiến trình cha và con.</li><li>• Trong tiến trình con, kiểm tra xem có định hướng file không? Nếu có thì dùng <b>dup2()</b> để chuyển hướng.</li><li>• Cho tiến trình con chạy lệnh bằng hàm <b>execvp()</b>.</li><li>• Tiến trình cha thì đợi nếu không có dấu '&amp;' trong lệnh.</li></ul>
<code>void execArgsPipe(char *args[], char *argsPipe[], int flag)</code>	Thực thi lệnh có dùng chức năng <b>pipe</b> . Ví dụ: <ul style="list-style-type: none"><li>• <b>osh&gt; ls -l   wc -l</b></li></ul>	<ul style="list-style-type: none"><li>• Dùng hàm <b>pipe()</b> để tạo ra một đường ống nối từ tiến trình cha đến tiến trình con.</li><li>• Dùng <b>fork()</b> để chia ra 2 tiến trình: ➔ Tiến trình con thực hiện lệnh phía tr " " → kết quả xuất vào</li></ul>

		<p>đường ống thông qua cổng <b>fd[1]</b> (lập trình viên tự đặt).</p> <p>➔ Tiến trình cha, tạo thêm 1 tiến trình con nữa. Tiến trình này sẽ nhận đầu vào từ <b>pipe</b> thông qua cổng <b>fd[0]</b>. Sau đó thực hiện lệnh phía sau “ ” . Còn tiến trình cha thì đợi cho đến khi tiến trình con thực hiện xong (<b>waitpid()</b>).</p>
<pre>void exec( char *args[], int len)</pre>	Thực thi cho một câu lệnh bất kì (kể cả trường hợp có dùng <b>pipe</b> ).	<p>Xét trường hợp có dấu “ ” bằng hàm <b>checkPipe()</b></p> <p>Nếu là hàm thông thường thì gọi hàm <b>execArgs()</b></p> <p>Nếu là hàm dùng <b>pipe</b> thì gọi hàm <b>execArgsPipe()</b></p>

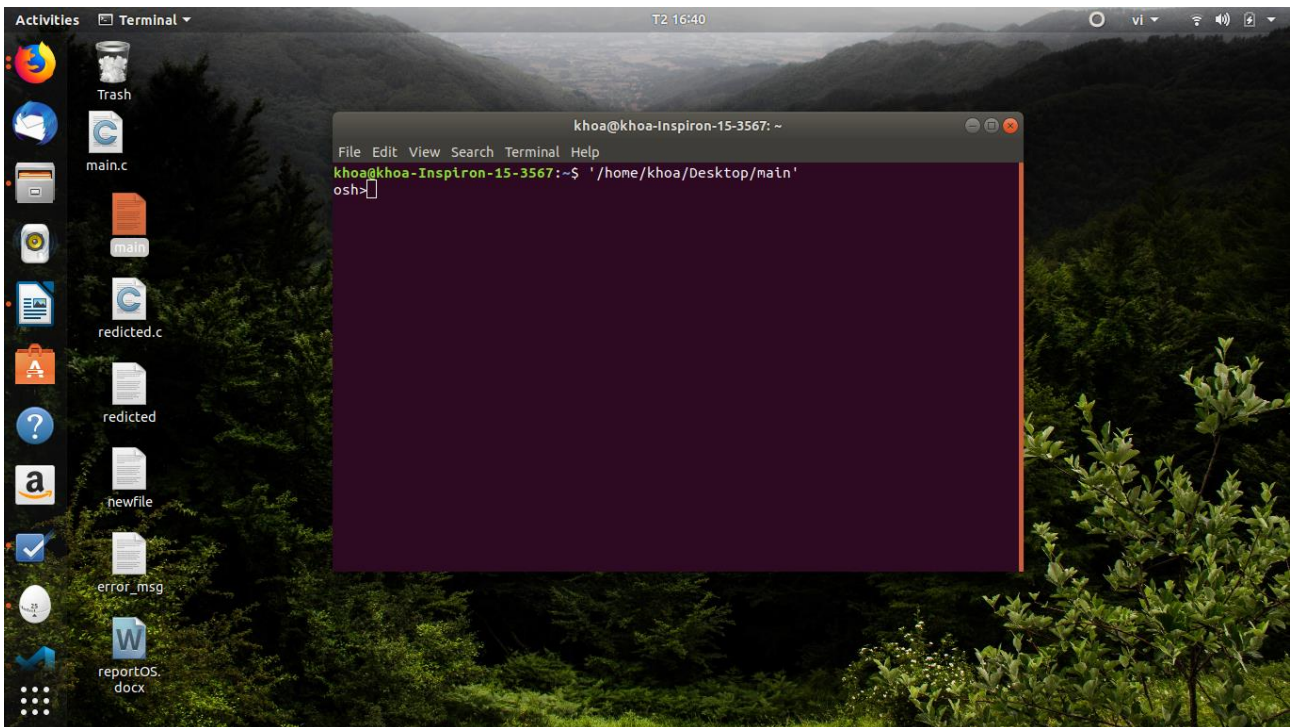
### III. Hướng dẫn cách chạy chương trình

**Lưu ý:** chương trình này chạy trên **Linux hoặc Ubuntu**, bạn có thể chạy máy ảo hoặc cài đặt Linux (Ubuntu) với Windows.

Bạn chạy bằng Visual Studio Code, bạn chỉ cần nhấp **F5** và chạy chương trình bình thường trong cửa sổ **Terminal**.



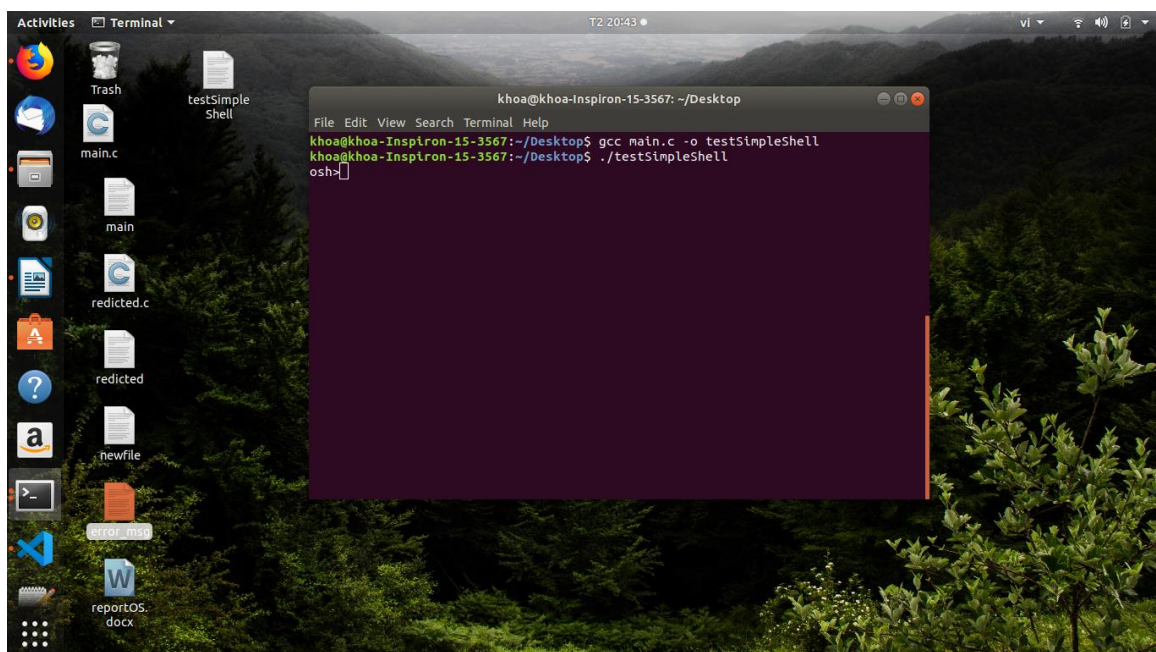
Hoặc kết hợp giữa VS Code với Terminal. Bạn nhấn **Ctrl + Shift + B** để build chương trình. Sau đó bạn nhấn **Ctrl + Alt + T** để mở Terminal lên. Cuối cùng bạn tìm file thực thi lúc này bạn vừa build (file này nằm trong thư mục **Files**) sau đó kéo thả nó vào **terminal**. Nhấn **Enter** để chạy chương trình.



Cách cuối cùng là bạn chỉ dùng Terminal. Bạn mở Terminal lên ( **Ctrl + Alt + T**). Gõ theo cú pháp biên dịch chương trình C như sau:

**\$ gcc [tên file .c] -o [tên file thực thi] → sau đó nhấn Enter**

**./[tên file thực thi] → sau đó nhấn Enter.**



## IV. Minh họa các yêu cầu trong đề án

### 1. Thực thi một lệnh thông thường

Thực thi lệnh `ls -l` để in ra danh sách các file hoặc thư mục trong thư mục hiện hành.

```
khoa@khoa-Inspiron-15-3567: ~/Desktop
File Edit View Search Terminal Help
osh>ls -l
total 1804
-rw-r--r-- 1 khoa khoa 154 Thg 1 7 11:55 error_msg
-rwxr-xr-x 1 khoa khoa 17632 Thg 1 7 20:29 main
-rw-rw-r-- 1 khoa khoa 5913 Thg 1 7 20:39 main.c
-rw-r--r-- 1 khoa khoa 24 Thg 1 7 11:51 newfile
-rwxr-xr-x 1 khoa khoa 11280 Thg 1 6 22:23 redicted
-rw-r--r-- 1 khoa khoa 717 Thg 1 6 22:21 redicted.c
-rw-rw-r-- 1 khoa khoa 1777066 Thg 1 7 16:59 reportOS.docx
-rwxr-xr-x 1 khoa khoa 13472 Thg 1 7 20:43 testSimpleShell
osh>
```

Lệnh “`cat`” dùng để in nội dung của một file đã tồn tại.

```
khoa@khoa-Inspiron-15-3567: ~/Desktop
File Edit View Search Terminal Help
osh>cat main.c
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <fcntl.h>
#include <string.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <sys/stat.h>

#define MAX_LINE 80

//Ham lay input
int getInput(char input[], char history[]){
    printf("osh>");
    fflush(stdout);
```

### 2. Sử dụng hàm thông thường với “&”

Khi dùng dấu ‘&’ thì cả tiến trình cha và tiến trình con đều cùng thực hiện nên “osh>” sẽ hiện ra trước vì nó tốn ít thời gian chạy hơn. Sau đó là danh sách chi tiết các thư mục hiện hành được in ra , bởi một tiến trình khác thực hiện.



```
khoa@khoa-Inspiron-15-3567: ~/Desktop
File Edit View Search Terminal Help
osh>ls -l &
osh>total 1804
-rw-r--r-- 1 khoa khoa    154 Thg 1  7 11:55 error_msg
-rwxr-xr-x 1 khoa khoa  17632 Thg 1  7 20:29 main
-rw-rw-r-- 1 khoa khoa   5913 Thg 1  7 20:39 main.c
-rw-r--r-- 1 khoa khoa     24 Thg 1  7 11:51 newfile
-rwxr-xr-x 1 khoa khoa  11280 Thg 1  6 22:23 redicted
-rw-r--r-- 1 khoa khoa    717 Thg 1  6 22:21 redicted.c
-rw-rw-r-- 1 khoa khoa 1777066 Thg 1  7 16:59 reportOS.docx
-rwxr-xr-x 1 khoa khoa   13472 Thg 1  7 20:43 testSimpleShell
```

### 3. Cung cấp lịch sử

Khi chưa có lệnh nào trong lịch sử

```
File Edit View Search Terminal Help
khoa@khoa-Inspiron-15-3567: ~
osh>!!
No command in history!!!
osh>
```

Nếu đã có lệnh trong lịch sử thì lệnh đó sẽ được in ra và thực hiện lệnh trong lịch sử

```
File Edit View Search Terminal Help
osh>ls
error_msg  main  main.c  newfile  redicted
osh>!!
ls
error_msg  main  main.c  newfile  redicted
osh>
```

### 4. Chuyển hướng input

Thay vì xuất ra màn hình, mình file (**newfile**)

```
khoa@khoa-Inspiron-15-3567: ~
File Edit View Search Terminal Help
osh>wc < newfile
1 3 12
osh>
```



## 5. Chuyển hướng output

```
File Edit View Search Terminal Help
osh>echo 'hello!!!' > newfile
osh>cat newfile
'hello!!!'
osh>
```

## 6. Chuyển hướng input có dấu '&'

```
File Edit View Search Terminal Help
osh>wc < newfile &
osh> 1 3 12
```

## 7. Chuyển hướng output có dấu '&'

```
File Edit View Search Terminal Help
osh>echo 'I love OS' > newfile &
osh>cat newfile
osh>'I love OS'
```

## 8. Sử dụng pipe()

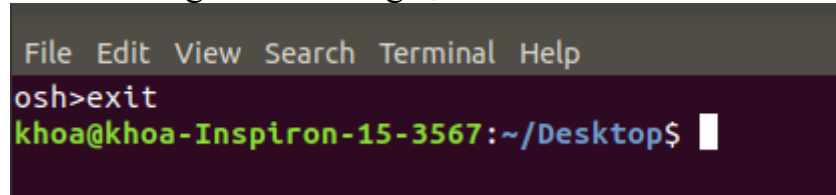
```
File Edit View Search Terminal Help
osh>printf 'line1\nline2\n' | wc -l
2
osh>
```

## 9. Sử dụng pipe() với dấu '&'

```
File Edit View Search Terminal Help
osh>ls -l | tail -4 &
osh>-rwxr-xr-x 1 khoa khoa 11280 Thg 1 6 22:23 redicted
-rw-r--r-- 1 khoa khoa 717 Thg 1 6 22:21 redicted.c
-rw-rw-r-- 1 khoa khoa 1777066 Thg 1 7 16:59 reportOS.docx
-rwxr-xr-x 1 khoa khoa 13472 Thg 1 7 20:55 testSimpleShell
```

## 10. Thoát khỏi chương trình

Để thoát khỏi chương trình ta dùng lệnh exit



```
File Edit View Search Terminal Help
osh>exit
khoa@khoa-Inspiron-15-3567:~/Desktop$
```

## V. Tham khảo

Tài liệu trên moodle Khoa

<http://www.cs.loyola.edu/~jglenn/702/S2005/Examples/dup2.html>

<https://viblo.asia/p/piping-va-chuyen-huong-cau-lenh-trong-linux-bJzKmk4OI9N>

<https://www.geeksforgeeks.org/pipe-system-call/>

<https://daynhauhoc.com/t/ham-fork-trong-he-dieu-hanh-linux-hoat-dong-nhu-the-nao/22893>

<http://www.csl.mtu.edu/cs4411.ck/www/NOTES/process/fork/create.html>