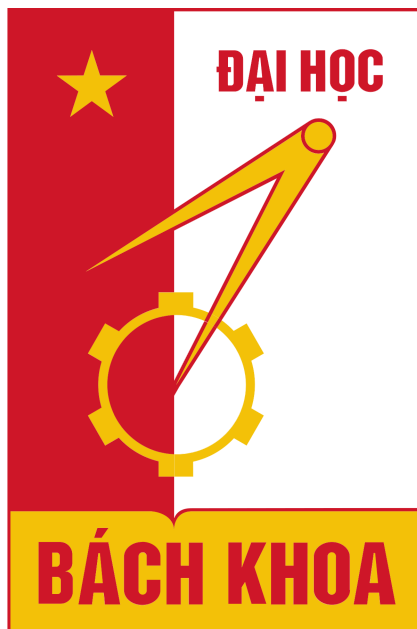


**ĐẠI HỌC BÁCH KHOA HÀ NỘI**  
**TRƯỜNG CÔNG NGHỆ THÔNG TIN VÀ TRUYỀN THÔNG**



**BÁO CÁO BÀI TẬP LỚN MÔN HỆ ĐIỀU HÀNH**  
**Đề tài: Tiny Shell**

Sinh viên thực hiện:	Trần Huy Hoàng
Mã số sinh viên:	20210386
Giảng viên hướng dẫn:	TS. Phạm Đăng Hải
Mã lớp:	143573

Hà Nội, tháng 1 năm 2024

## Mục lục

<b>1. Giới thiệu tổng quát về Tiny Shell.....</b>	<b>3</b>
<b>2. Giới thiệu các chức năng quản lý tiến trình.....</b>	<b>4</b>
2.1. Thực hiện tiến trình ở chế độ foreground, background.....	4
2.2. Hiển thị danh sách các tiến trình.....	5
2.3. Kill một/tất cả background process.....	6
2.4. Stop và Resume một background process.....	7
2.5. Hủy bỏ foreground process bằng tín hiệu ngắt (CTRL+C).....	8
<b>3. Giới thiệu các chức năng khác.....</b>	<b>9</b>
3.1. Chức năng của lệnh help.....	9
3.2. Chức năng của lệnh dir.....	10
3.3. Chức năng của lệnh clear và exit.....	10
3.4. Chức năng của lệnh date và time.....	10
3.5. Chức năng của lệnh cd.....	11
3.6. Chức năng của lệnh path và addpath.....	12
3.7. Chức năng của lệnh exec.....	13
<b>4. Tổng kết.....</b>	<b>14</b>

## 1. Giới thiệu tổng quát về Tiny Shell

Tiny Shell là một giao diện dòng lệnh (command-line interface) giúp người dùng tương tác với hệ điều hành Linux bằng cách nhập lệnh và nhận kết quả trả về. Tiny Shell có thể thực hiện được các chức năng như sau:

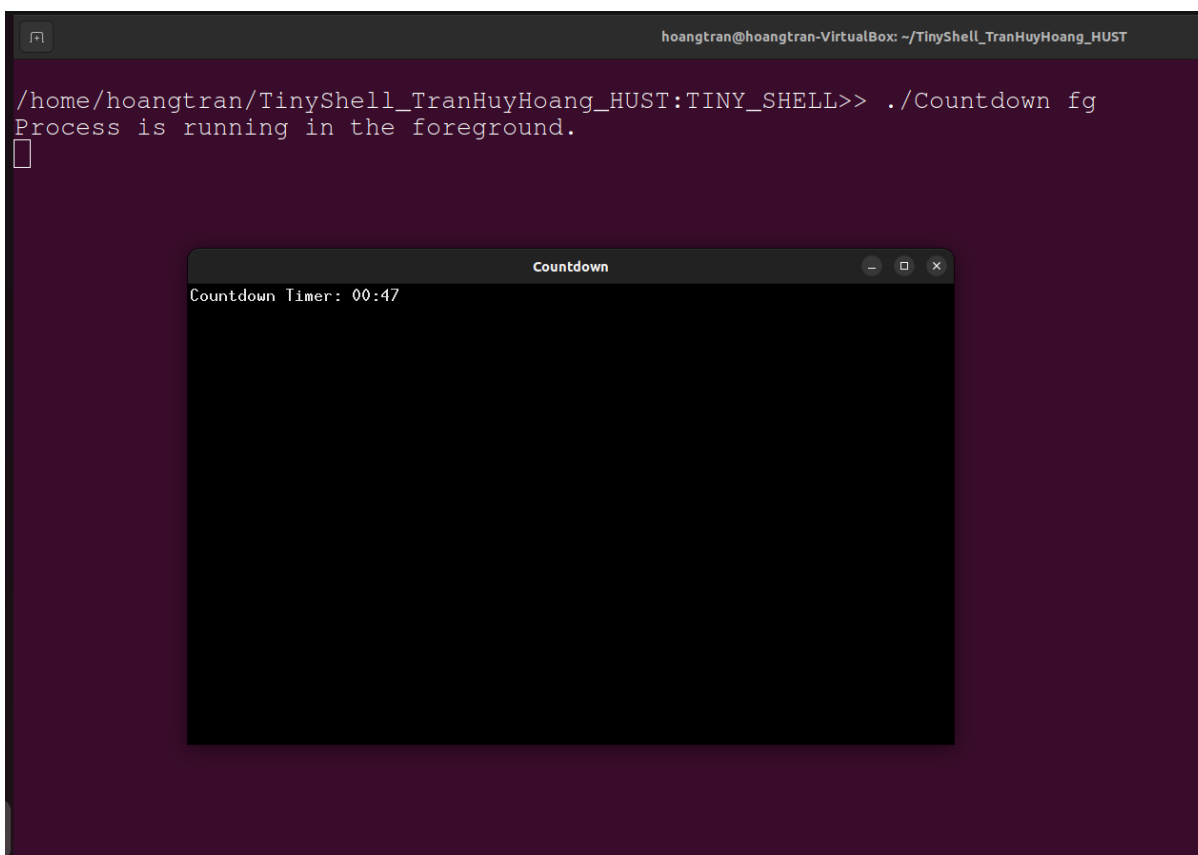
STT	Tên câu lệnh	Chức năng
1	help	Giải thích ý nghĩa của tất cả các câu lệnh
2	dir	Liệt kê các tệp và thư mục con của thư mục hiện tại
3	clear	Xóa màn hình Tiny Shell
4	exit	Thoát Tiny Shell
5	date	Hiển thị ngày hiện tại
6	time	Hiển thị giờ hiện tại
7	cd [path]	Thay đổi thư mục làm việc hiện tại thành thư mục có đường dẫn [path]
8	path	Hiển thị biến môi trường PATH
9	addpath [path]	Thêm đường dẫn [path] vào biến môi trường PATH
10	[path] fg	Chạy tiến trình có đường dẫn [path] ở chế độ foreground
11	[path] bg	Chạy tiến trình có đường dẫn [path] ở chế độ background
12	list	Hiển thị danh sách các tiến trình
13	kill [id]	Hủy một background process có ID là [id]
14	killa	Hủy tất cả background process đang chạy
15	stop [id]	Tạm dừng một background process có ID là [id]
16	resume [id]	Tiếp tục thực hiện một background process đang tạm dừng có ID là [id]
17	<CTRL + C>	Hủy bỏ foreground process bằng CTRL+C
18	exec [path]	Thực thi file *.sh có đường dẫn là [path]

## 2. Giới thiệu các chức năng quản lý tiến trình

### 2.1. Thực hiện tiến trình ở chế độ foreground, background

Để thực hiện tiến trình ở chế độ foreground, ta nhập lệnh sau: **[path] fg**  
Trong đó **[path]** là đường dẫn đến file thực thi của tiến trình cần chạy.

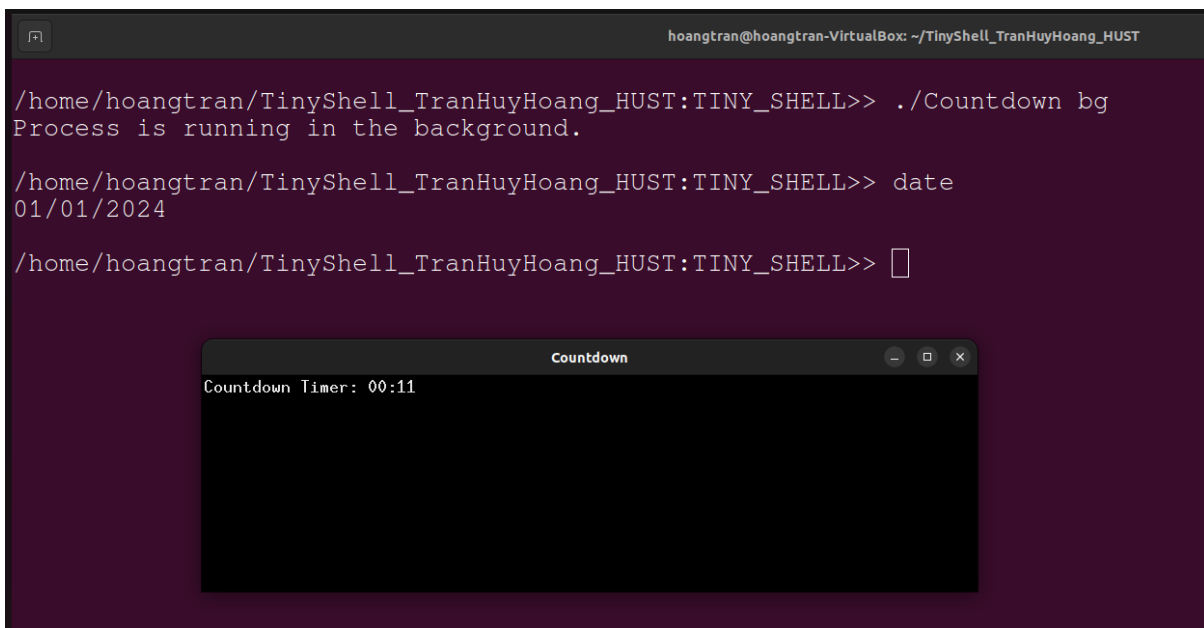
Ví dụ: trong thư mục làm việc hiện tại có file thực thi Countdown (Countdown là chương trình đồng hồ đếm ngược). Khi nhập lệnh **./Countdown fg** vào Tiny Shell, ta được kết quả như sau:



Ở chế độ foreground, Tiny Shell phải đợi tiến trình con kết thúc mới có thể tiếp tục nhận lệnh.

Tương tự, để thực hiện tiến trình ở chế độ background, ta nhập lệnh sau: **[path] bg**

Ví dụ : Khi nhập lệnh **./Countdown bg**, ta được kết quả như sau:



```
hoangtran@hoangtran-VirtualBox: ~/TinyShell_TranHuyHoang_HUST
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> ./Countdown bg
Process is running in the background.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> date
01/01/2024

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> 
```

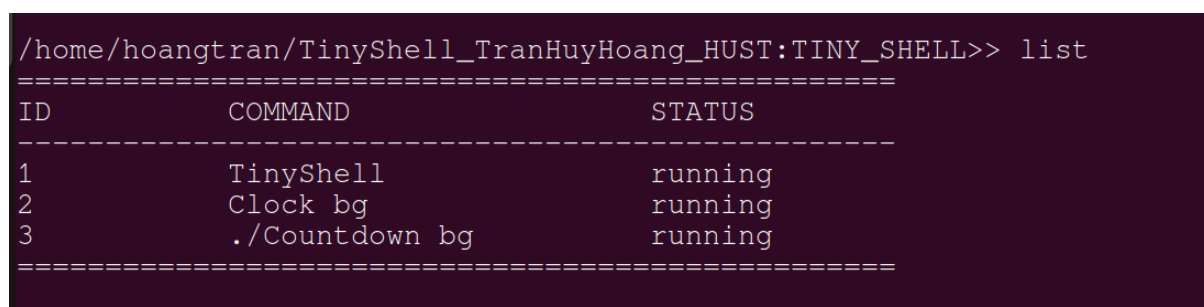
The screenshot shows a terminal window with a dark purple background. The prompt is `hoangtran@hoangtran-VirtualBox: ~/TinyShell_TranHuyHoang_HUST`. The user enters `./Countdown bg`, and the output is `Process is running in the background.`. Then the user enters `date`, and the output is `01/01/2024`. Finally, the user enters a blank line. Overlaid on the terminal is a small window titled "Countdown" with a black background and white text that reads "Countdown Timer: 00:11".

Ở chế độ background, Tiny Shell và tiến trình con có thể thực hiện song song với nhau.

## 2.2. Hiển thị danh sách các tiến trình

Khi dùng câu lệnh **list**, Tiny Shell sẽ hiển thị danh sách các tiến trình, bao gồm các thông tin : process ID, command name, status.

Kết quả minh họa khi chạy lệnh **list** như sau:



```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell         running
2           Clock bg          running
3           ./Countdown bg    running
=====
```

The screenshot shows a terminal window with a dark purple background. The prompt is `/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>>`. The user enters `list`, and the output is a table with three columns: ID, COMMAND, and STATUS. The table is separated by dashed lines. The data rows are: 1, TinyShell, running; 2, Clock bg, running; 3, ./Countdown bg, running. The table is preceded and followed by lines of equals signs.

## 2.3. Kill một/tất cả background process

### Kill một background process đang chạy

Để kill một background process đang chạy, ta dùng lệnh: **kill [id]**, trong đó **[id]** là id của background process cần kill.

Kết quả minh họa khi chạy lệnh **kill [id]** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> ./Countdown bg
Process is running in the background.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell         running
2           ./Countdown bg    running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> kill 2
Process with ID 2 has been terminated.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell         running
=====
```

Giải thích kết quả minh họa: Khi chạy lệnh **./Countdown bg** thì tiến trình này được thêm vào bảng quản lý tiến trình với ID bằng 2. Khi dùng lệnh **kill 2** thì tiến trình này sẽ kết thúc và sẽ không còn xuất hiện trong bảng quản lý tiến trình.

Ta cũng có thể dùng lệnh **kill 1** để kết thúc chính Tiny Shell:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell         running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> kill 1
Exiting TinyShell. Goodbye!
hoangtran@hoangtran-VirtualBox:~/TinyShell_TranHuyHoang_HUST$
```

## Kill tất cả các background process đang chạy

Để kill tất cả các background process đang chạy, ta dùng lệnh **killall**.

Kết quả minh họa khi chạy lệnh **killall** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell          running
2           Clock bg           running
3           ./Countdown bg     running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> killall
All processes have been terminated.

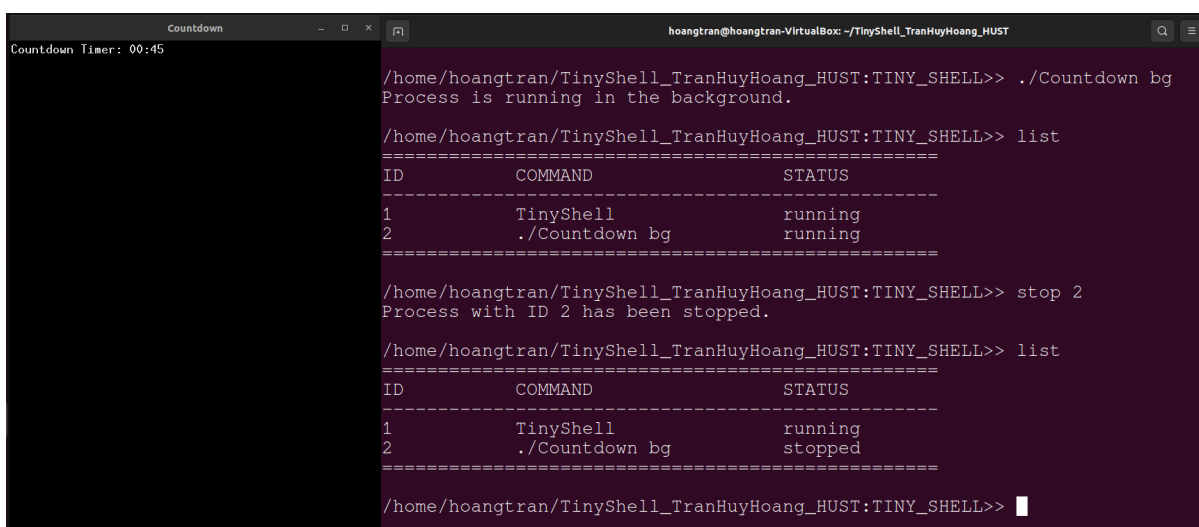
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell          running
=====
```

## 2.4. Stop và Resume một background process

### Stop một background process đang chạy

Khi ta dùng lệnh **stop [id]** thì background process đang chạy có ID bằng **[id]** sẽ bị tạm dừng. Lúc này, status của background process này sẽ chuyển từ running sang stopped.

Kết quả minh họa khi chạy lệnh **stop [id]** như sau:



The screenshot shows a terminal window with a dark background. On the left, there is a window titled 'Countdown' with a timer at 00:45. The main terminal window shows the following commands and output:

```
hoangtran@hoangtran-VirtualBox: ~/TinyShell_TranHuyHoang_HUST

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> ./Countdown bg
Process is running in the background.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell          running
2           ./Countdown bg     running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> stop 2
Process with ID 2 has been stopped.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID          COMMAND          STATUS
-----
1           TinyShell          running
2           ./Countdown bg     stopped
=====

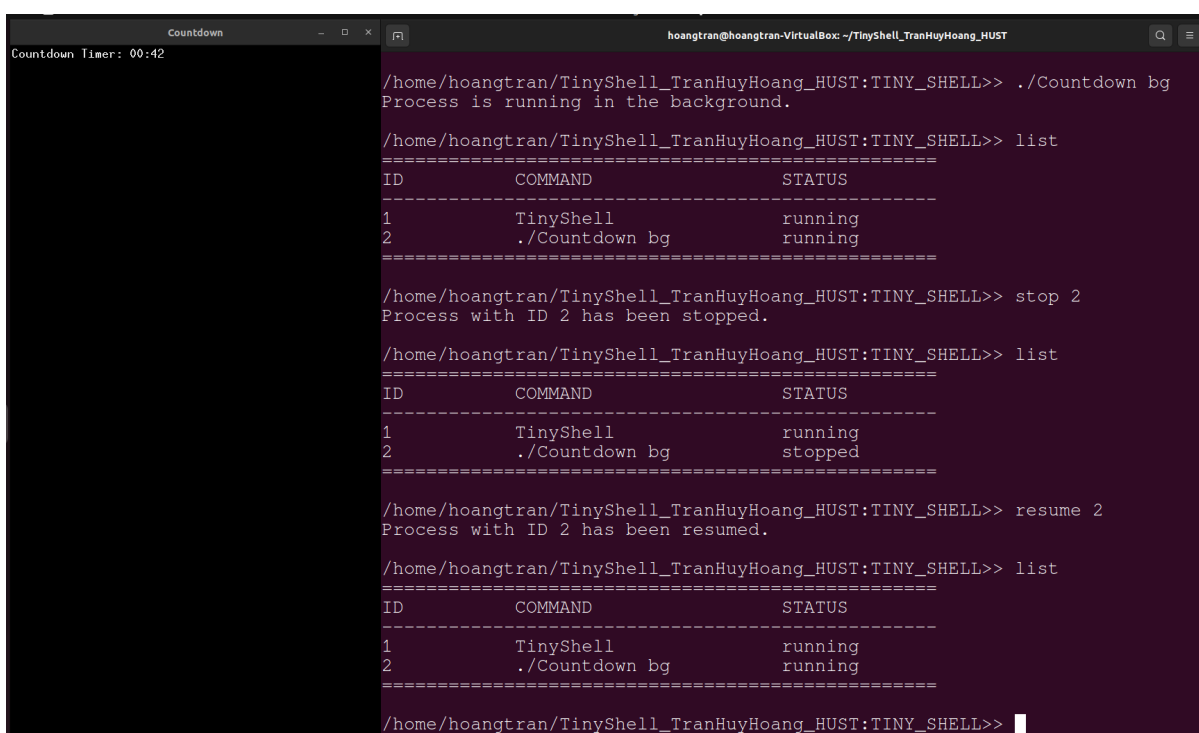
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> 
```

Giải thích kết quả minh họa: ta dùng lệnh **./Countdown bg** để chạy tiến trình này ở chế độ background. Lúc này, tiến trình này có ID là 2 và status là running. Khi ta dùng lệnh **stop 2**, tiến trình này sẽ bị tạm dừng (Countdown timer tạm dừng ở 00:45) và status sẽ trở thành stopped.

### Resume một background process đang chạy

Khi ta dùng lệnh **resume [id]** thì background process đang tạm dừng (stopped) có ID bằng **[id]** sẽ được tiếp tục thực hiện. Lúc này, status của background process này sẽ chuyển từ stopped sang running.

Kết quả minh họa khi chạy lệnh **resume [id]** như sau:



```
Countdown Timer: 00:42

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> ./Countdown bg
Process is running in the background.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID      COMMAND      STATUS
-----
1       TinyShell    running
2       ./Countdown bg running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> stop 2
Process with ID 2 has been stopped.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID      COMMAND      STATUS
-----
1       TinyShell    running
2       ./Countdown bg stopped
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> resume 2
Process with ID 2 has been resumed.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> list
=====
ID      COMMAND      STATUS
-----
1       TinyShell    running
2       ./Countdown bg running
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> 
```

Giải thích kết quả minh họa: Sau lệnh **stop 2**, background process **./Countdown bg** có status là stopped (Countdown timer đang tạm dừng ở 00:45). Khi ta dùng lệnh **resume 2**, tiến trình này sẽ tiếp tục thực hiện và chuyển status từ stopped sang running (Countdown timer tiếp tục đếm ngược từ 00:45)

## 2.5. Hủy bỏ foreground process bằng tín hiệu ngắt (CTRL+C)

Khi một foreground process đang chạy, ta có thể sử dụng tổ hợp phím **CTRL+C** để hủy foreground process đó



Kết quả minh họa khi dùng **CTRL+C** để hủy tiến trình **./Countdown fg**:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> ./Countdown fg
Process is running in the foreground.
^C
Ctrl+C pressed. Foreground process has terminated

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> █
```

### 3. Giới thiệu các chức năng khác

#### 3.1. Chức năng của lệnh help

Khi dùng lệnh **help**, Tiny Shell sẽ in ra danh sách tất cả câu lệnh, bao gồm No (số thứ tự), Command (tên câu lệnh), Function (chức năng của câu lệnh). Thông báo “*Bad command! Please try again or type 'help' to see a list of commands*” sẽ hiện ra giúp người dùng biết sử dụng **help** mỗi khi người dùng nhập sai lệnh.

Kết quả khi dùng lệnh **help** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> abcdef
Bad command! Please try again or type 'help' to see a list of commands.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> help
Tiny Shell - Tran Huy Hoang 20210386 - HUST
LIST OF COMMAND
=====
No          Command      Function
-----
1           help         Explain the meaning of all commands
2           dir           List files and subfolders in the current directory
3           clear        Clear Tiny Shell screen
4           exit         Exit Tiny Shell
5           date         Displays the current date
6           time         Displays the current time
7           cd [path]    Change the current working directory to the directory with path [path]
8           path         Displays the PATH environment variable
9           addpath [path] Add [path] to the PATH environment variable
10          [path] fg    Run process with path [path] in foreground mode
11          [path] bg    Run process with path [path] in background mode
12          list        Displays a list of processes
13          kill [id]    Kill a background process with ID [id]
14          killa       Kill all running background processes
15          stop [id]    Stop a background process with ID [id]
16          resume [id] Resume execution of a paused background process with ID [id]
17          <CTRL + C> Terminate the foreground process using the key combination CTRL + C
18          exec [path] Execute the *.sh file whose path is [path]
=====

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>>
```

### 3.2. Chức năng của lệnh **dir**

Lệnh **dir** dùng để liệt kê các tệp và thư mục con ở trong thư mục làm việc hiện tại. Ngoài các tên tệp và thư mục thông thường thì còn có thư mục “.” và thư mục “..”. Thư mục “.” đại diện cho chính thư mục làm việc hiện tại, còn thư mục “..” là thư mục cha của thư mục hiện tại.

Kết quả khi dùng lệnh **dir** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> dir
CommandLib.h      .vscode .git      TinyShell.c      .Xresources      README.md      .
..                Countdown  PrintNum.c      script.sh        PrintNum
ProcessLib.h      TinyShell  Countdown.c
```

### 3.3. Chức năng của lệnh **clear** và **exit**

Lệnh **clear** dùng để xóa toàn bộ nội dung trên màn hình của Tiny Shell

Kết quả khi dùng lệnh **clear** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> 
```

Lệnh **exit** dùng để thoát khỏi Tiny Shell

Kết quả khi dùng lệnh **exit** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> exit
Exit TinyShell... Goodbye!
hoangtran@hoangtran-VirtualBox:~/TinyShell_TranHuyHoang_HUST$
```

### 3.4. Chức năng của lệnh **date** và **time**

Khi dùng lệnh **date**, Tiny Shell sẽ hiển thị ngày hiện tại với định dạng dd/mm/yyyy

Kết quả khi dùng lệnh **date** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> date
02/01/2024

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>>
```

Khi dùng lệnh **time**, Tiny Shell sẽ hiển thị giờ hiện tại với định dạng hh:mm:ss

Kết quả khi dùng lệnh **time** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> time
01:03:07

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>>
```

### 3.5. Chức năng của lệnh **cd**

Khi chúng ta dùng lệnh **cd [path]** thì thư mục làm việc hiện tại sẽ được thay đổi thành thư mục có đường dẫn là **[path]**, trong đó **[path]** là đường dẫn tương đối hoặc đường dẫn tuyệt đối.

Kết quả khi dùng lệnh **cd [path]** với đường dẫn tương đối như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> cd ../Downloads
Current directory has been changed!

/home/hoangtran/Downloads:TINY_SHELL>>
```

Kết quả khi dùng lệnh **cd [path]** với đường dẫn tuyệt đối như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> cd /home/hoangtran/Downloads
Current directory has been changed!

/home/hoangtran/Downloads:TINY_SHELL>> █
```

### 3.6. Chức năng của lệnh **path** và **addpath**

Biến **PATH** là biến môi trường chứa danh sách các đường dẫn mà Linux sẽ tìm kiếm các tệp thực thi khi chạy lệnh.

Lệnh **path** dùng để in ra giá trị của biến môi trường **PATH**.

Kết quả khi chạy lệnh **path** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> path
PATH = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> █
```

Lệnh **addpath [path]** dùng để thêm đường dẫn tuyệt đối **[path]** vào biến môi trường **PATH**.

Kết quả khi chạy lệnh **addpath [path]** như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> addpath /home/hoangtran/newPath
Add new path success!
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> █
```

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> path
PATH = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin:/home/hoangtran/newPath

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>>
```

Giải thích kết quả: khi dùng lệnh **addpath /home/hoangtran/newPath**, biến **PATH** đã trở thành **/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin:/home/hoangtran/newPath** (thêm đường dẫn **/home/hoangtran/newPath** so với biến **PATH** trước khi thực hiện lệnh **addpath**).

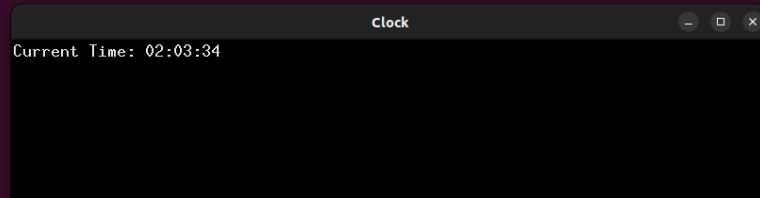
Sau khi chạy lệnh **addpath /home/hoangtran/newPath**, chúng ta thử thực hiện lệnh **Clock bg** (**Clock** là file có thể thực thi được ở trong **/home/hoangtran/newPath**). Ta thấy chương trình chạy bình thường mặc dù trong thư mục hiện tại không có file **Clock**. Điều này chứng tỏ **/home/hoangtran/newPath** đã được thêm vào biến **PATH** thành công.

Kết quả khi thực hiện lệnh **Clock bg** sau khi thực hiện **addpath**:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> path
PATH = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/sbin:/bin:/usr/games:/usr/local/games:/snap/bin:/snap/bin:/home/hoangtran/newPath

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> Clock bg
Process is running in the background.

/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> □
```



### 3.7. Chức năng của lệnh **exec**

Shell script là một tập hợp các lệnh dòng lệnh được sắp xếp theo một trình tự cụ thể và được lưu trong một tệp tin văn bản có đuôi là **.sh**.

Lệnh **exec [path]** dùng để thực thi file **.sh** có đường dẫn là **[path]**.

Ví dụ minh họa khi chạy lệnh **exec [path]**:

Trong thư mục làm việc hiện tại có file **script.sh** với nội dung như sau:

```
dir
time
date
```

Khi chạy lệnh **exec ./script.sh**, ta được kết quả như sau:

```
/home/hoangtran/TinyShell_TranHuyHoang_HUST:TINY_SHELL>> exec ./script.sh
CommandLib.h      .vscode .git      TinyShell.c      .Xresources      README.md
...              Countdown  script.sh.save  PrintNum.c      script.sh
PrintNumProcessLib.h TinyShell    Countdown.c
02:50:00
02/01/2024
```

Giải thích kết quả: khi thực hiện lệnh **exec ./script.sh**, Tiny Shell sẽ lần lượt thực hiện các câu lệnh có trong nội dung file **script.sh**, đó là các lệnh **dir**, **time**, **date**.

#### **4. Tổng kết**

Tiny Shell là một phiên bản đơn giản của Shell trong Linux, thông qua quá trình phát triển Tiny Shell, em đã học thêm được nhiều kiến thức về cách quản lý tiến trình cũng như cách thức Shell trong hệ điều hành Linux hoạt động.

Em xin chân thành cảm ơn Thầy Phạm Đăng Hải đã giảng dạy những kiến thức hữu ích của môn Nguyên lý hệ điều hành để em thể phát triển được project này.

Trong quá trình phát triển Tiny Shell, dù em đã cố gắng hoàn thiện nhất có thể, tuy nhiên không thể tránh khỏi những sai sót không đáng có. Vì vậy em rất mong nhận được những lời khuyên và sự góp ý từ Thầy để em có thể tiếp tục phát triển trong tương lai.