

Simple Shell (Multiprocessing)

Yara Hossam Abdelaziz 19016871

Problem Statement:

Implement a Unix shell program with the following commands:

1. exit
2. Commands with no arguments: ls, cp, rm, ..etc
3. Commands with arguments: ls -a -l -h
4. Background commands
5. Shell builtin commands: cd, echo
6. Expression evaluation: export

Code Organization & Main Functions:

1. set_up_environment(): open log file and initialize signal connection to be called on child exit and prevent zombie processes.
2. parse_input() : takes user input and splits it at the space to the command and arguments.
3. shell(): forks a child process using fork(), executes the command using execvp() and directs the parent process to wait on the child

process using `waitpid()` till it executes successfully if it's a foreground process.

4. `cd()`: changes directory using `chdir` to a given path or home.
5. `export()`: evaluates an expression/value to an environment variable.
6. `echo()`: prints given argument/expression.

Sample Runs:

1. Basic commands:

```
yara@yara:~ ls
logFile.txt  shell  shell.c  tempCodeRunnerFile.c  test  test2  test3

yara@yara:~ mkdir yara

yara@yara:~ ls
logFile.txt  shell  shell.c  tempCodeRunnerFile.c  test  test2  test3  yara
```

2. Evaluating expressions:

```
yara@yara:~ export x=-l

yara@yara:~ ls $x
total 48
-rw-rw-r-- 1 yara yara  520 Mar 24 05:16 logFile.txt
-rwxrwxr-x 1 yara yara 13824 Mar 24 05:15 shell
-rw-rw-r-- 1 yara yara  4297 Mar 24 05:14 shell.c
-rw-rw-r-- 1 yara yara    6 Mar 24 04:12 tempCodeRunnerFile.c
drwxrwxr-x 2 yara yara  4096 Mar 24 01:13 test
drwxrwxr-x 2 yara yara  4096 Mar 24 01:14 test2
drwxrwxr-x 2 yara yara  4096 Mar 24 01:15 test3
drwxrwxr-x 2 yara yara  4096 Mar 24 05:15 yara
```

3.

```
yara@yara:~ export y="Hello World"

yara@yara:~ echo $y

Hello World
yara@yara:~ hello

Execution failed.
yara@yara:~
```

4.

```
yara@yara:~ export q="-a -l -h"

yara@yara:~ ls $q
total 56K
drwxrwxr-x 6 yara yara 4.0K Mar 24 05:15 .
drwxrwxr-x 3 yara yara 4.0K Mar 24 00:33 ..
-rw-rw-r-- 1 yara yara 572 Mar 24 05:19 logFile.txt
-rwxrwxr-x 1 yara yara 14K Mar 24 05:15 shell
-rw-rw-r-- 1 yara yara 4.2K Mar 24 05:14 shell.c
-rw-rw-r-- 1 yara yara 6 Mar 24 04:12 tempCodeRunnerFile.c
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:13 test
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:14 test2
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:15 test3
drwxrwxr-x 2 yara yara 4.0K Mar 24 05:15 yara

yara@yara:~
```

5. Basic command to make a copy of a file then remove it:

```
yara@yara:~ cp logFile.txt a.txt  
  
yara@yara:~ ls  
a.txt logFile.txt shell shell.c tempCodeRunnerFile.c test test2 test3 yara  
  
yara@yara:~ rm a.txt  
  
yara@yara:~ ls  
logFile.txt shell shell.c tempCodeRunnerFile.c test test2 test3 yara  
  
yara@yara:~ exit  
yara@yara:~/Desktop/boo/Shell$
```

6. Opening firefox in the foreground:

```
yara@yara:~ firefox  
ls  
mkdir os  
█
```

7. Closing firefox and the commands get executed by then:

```
yara@yara:~ firefox  
ls  
mkdir os  
  
yara@yara:~ logFile.txt shell shell.c tempCodeRunnerFile.c test test2 test3 yar  
  
yara@yara:~  
yara@yara:~ ls  
logFile.txt os shell shell.c tempCodeRunnerFile.c test test2 test3 yara  
  
yara@yara:~ █
```

8. Opening firefox in the background with the commands executing normally:

```
cd "/home/yara/Desktop/boo/Shell/" && gcc shell.c -o shell &&
yara@yara:~/Desktop/boo$ cd "/home/yara/Desktop/boo/Shell/" &
yara@yara:~$ firefox &
yara@yara:~$ ls
logFile.txt  os  shell  shell.c  tempCodeRunnerFile.c  test
yara@yara:~$ ls -a -l -h
total 60K
drwxrwxr-x 7 yara yara 4.0K Mar 24 05:37 .
drwxrwxr-x 3 yara yara 4.0K Mar 24 00:33 ..
-rw-rw-r-- 1 yara yara 988 Mar 24 05:37 logFile.txt
drwxrwxr-x 2 yara yara 4.0K Mar 24 05:34 os
-rwxrwxr-x 1 yara yara 14K Mar 24 05:37 shell
-rw-rw-r-- 1 yara yara 4.2K Mar 24 05:22 shell.c
-rw-rw-r-- 1 yara yara 6 Mar 24 04:12 tempCodeRunnerFile.c
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:13 test
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:14 test2
drwxrwxr-x 2 yara yara 4.0K Mar 24 01:15 test3
drwxrwxr-x 2 yara yara 4.0K Mar 24 05:15 yara
yara@yara:~$
```

Screenshots of the System Monitor during execution:

Open firefox in the foreground:

The screenshot shows the System Monitor application with the 'Processes' tab selected. A search bar at the top contains the text 'firefox'. The process list shows a hierarchy starting with 'code --no-sandbox --force-use', which includes a 'code' process, which in turn includes a 'bash' process, which includes a 'shell' process, which finally includes the 'firefox' process. The 'firefox' process is highlighted, showing its details: MainThread (11129, 10.7 MB), Privileged Cont (10983, 26.9 MB), Socket Process (10956, 7.5 MB), Web Content (11114, 10.8 MB), Web Content (11105, 10.7 MB), and WebExtensions (11025, 19.2 MB). On the left, a terminal window shows the commands: `cd "/home/yara/Desktop/boo/Shell/"`, `yara@yara:~/Desktop/boo$ cd "/home/yara/Desktop/boo/Shell/"`, `yara@yara:~/Desktop/boo$ firefox`, `ls`, and `mkdir os`.

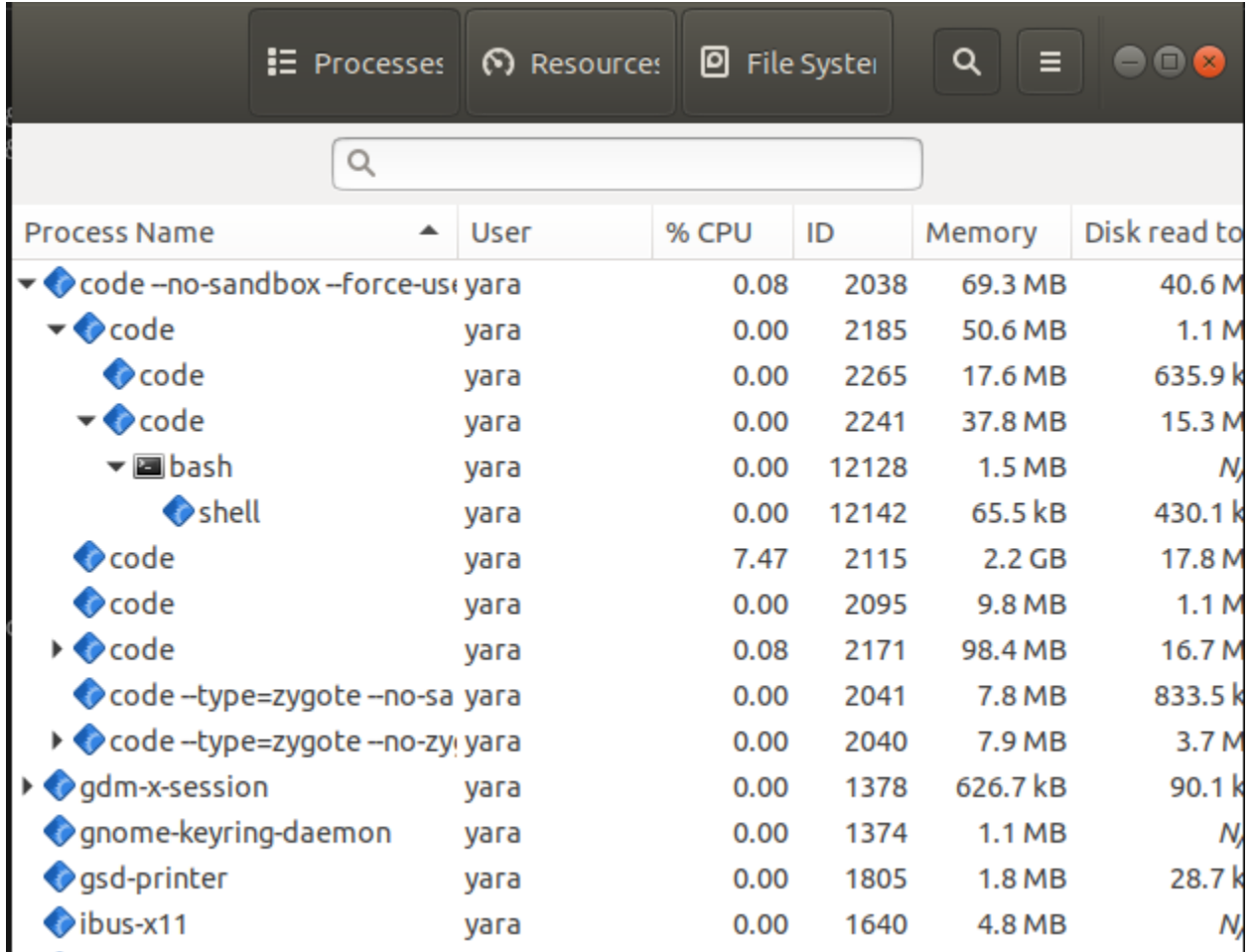
Process Name	User	% CPU	ID	Memory	Disk read to
code --no-sandbox --force-use	yara	0.25	2038	68.6 MB	40.6 M
code	yara	0.00	2185	50.3 MB	1.1 M
code	yara	0.00	2241	37.9 MB	15.3 M
bash	yara	0.00	10865	1.5 MB	N
shell	yara	0.00	10879	65.5 kB	N
firefox	yara	0.00	10896	114.6 MB	N
MainThread	yara	0.00	11129	10.7 MB	N
Privileged Cont	yara	0.00	10983	26.9 MB	N
Socket Process	yara	0.00	10956	7.5 MB	N
Web Content	yara	0.00	11114	10.8 MB	N
Web Content	yara	0.00	11105	10.7 MB	N
WebExtensions	yara	0.00	11025	19.2 MB	N

Open firefox in the background:

The screenshot shows the System Monitor application with the 'Processes' tab selected. A search bar at the top is empty. The process list shows a hierarchy starting with 'code --no-sandbox --force-use', which includes a 'code' process, which includes a 'code' process, which includes a 'bash' process, which includes a 'shell' process, which finally includes the 'firefox' process. The 'firefox' process is highlighted, showing its details: Privileged Cont (12251, 27.5 MB), Socket Process (12217, 7.5 MB), Web Content (12381, 10.5 MB), Web Content (12384, 10.5 MB), Web Content (12390, 10.5 MB), and WebExtensions (12297, 20.2 MB). On the left, a terminal window shows the commands: `cd "/home/yara/Desktop/boo/Shell/"`, `yara@yara:~/Desktop/boo$ cd "/home/yara/Desktop/boo/Shell/"`, `yara@yara:~/Desktop/boo$ firefox &`, `yara@yara:~/Desktop/boo$ ls`, `logFile.txt`, `os`, `shell`, `shell.c`, `tempCodeRunnerFile.c`, `test`, `yara@yara:~/Desktop/boo$ ls -a -l -h`, and `yara@yara:~/Desktop/boo$`. The output of the `ls -a -l -h` command is shown, listing files and their permissions, sizes, and timestamps.

Process Name	User	% CPU	ID	Memory	Disk read to
code --no-sandbox --force-use	yara	0.08	2038	69.3 MB	40.6 M
code	yara	0.17	2185	50.4 MB	1.1 M
code	yara	0.00	2265	17.6 MB	635.9 k
code	yara	0.00	2241	37.8 MB	15.3 M
bash	yara	0.00	12128	1.5 MB	N
shell	yara	0.00	12142	65.5 kB	N
firefox	yara	0.08	12160	115.3 MB	N
Privileged Cont	yara	0.00	12251	27.5 MB	N
Socket Process	yara	0.00	12217	7.5 MB	N
Web Content	yara	0.00	12381	10.5 MB	N
Web Content	yara	0.00	12384	10.5 MB	N
Web Content	yara	0.00	12390	10.5 MB	N
WebExtensions	yara	0.00	12297	20.2 MB	N
code	yara	4.36	2115	2.2 GB	17.8 M
code	yara	0.00	2095	9.8 MB	1.1 M

When exiting firefox:



<div>Processes Resources File System</div> <div></div>						
<div></div>						
Process Name	User	% CPU	ID	Memory	Disk read to	
code --no-sandbox --force-use	yara	0.08	2038	69.3 MB	40.6 M	
code	yara	0.00	2185	50.6 MB	1.1 M	
code	yara	0.00	2265	17.6 MB	635.9 k	
code	yara	0.00	2241	37.8 MB	15.3 M	
bash	yara	0.00	12128	1.5 MB	N/	
shell	yara	0.00	12142	65.5 kB	430.1 k	
code	yara	7.47	2115	2.2 GB	17.8 M	
code	yara	0.00	2095	9.8 MB	1.1 M	
code	yara	0.08	2171	98.4 MB	16.7 M	
code --type=zygote --no-sa	yara	0.00	2041	7.8 MB	833.5 k	
code --type=zygote --no-zy	yara	0.00	2040	7.9 MB	3.7 M	
gdm-x-session	yara	0.00	1378	626.7 kB	90.1 k	
gnome-keyring-daemon	yara	0.00	1374	1.1 MB	N/	
gsd-printer	yara	0.00	1805	1.8 MB	28.7 k	
ibus-x11	yara	0.00	1640	4.8 MB	N/	