

## [2016 Network System Programming Homework2]

This homework focuses on system programming and pipe.

### Part 1:

1. Write, compile, and run a program named **hostinfo** that prints out system information in the following format.

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./hostinfo
hostname: jth-VirtualBox
Linux 4.4.0-31-generic
hostid: 8323329
```

2. Write, compile, and run a program named **mydate** that prints out the day and time in the following format.

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./mydate
Oct 3(Mon), 2016 6:52 PM
```

3. Write a program called **printdir** that prints the current directory. Determine what size buffer to pass to **getcwd()** for dynamic allocation. (Do not use **pwd()**.)

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./printdir
/home/jth/Desktop/M043040098_SP_HW2/part1
```

4. Write a program called **mycat** that is a simple version of the program **cat**. The program takes exactly one file name as argument; you should open it for reading and display its contents to the screen. Check that there is exactly one argument (**argc == 2**) and if not, display the usage message "**Usage: mycat filename**".

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ cat 123
123456
ABCDE
***
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./mycat 123
123456
ABCDE
***
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./mycat 123 456
Usage: mycat filename
```

5. Write **pipe\_ls** to practice using **pipe()** and **dup()**. Have your process start **ls** (using **fork()** and **exec()**) but read the output from **ls** over a pipe. The **ls** program writes output on descriptor 1, so some work has to be done to get the pipe connected. Write what you read on the pipe to **stdout**.

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part1$ ./pipe_ls
total 96
drwxrwxr-x 2 jth jth 4096 + 3 19:48 .
drwxrwxr-x 5 jth jth 4096 + 3 18:41 ..
-rw-rw-r-- 1 jth jth 17 + 3 19:48 123
-rwxrwxr-x 1 jth jth 8816 + 3 19:48 hostinfo
-rw-rw-r-- 1 jth jth 352 + 3 19:47 hostinfo.c
-rw-rw-r-- 1 jth jth 189 + 3 19:47 makefile
-rwxrwxr-x 1 jth jth 8912 + 3 19:48 mycat
-rw-rw-r-- 1 jth jth 363 + 3 18:48 mycat.c
-rwxrwxr-x 1 jth jth 8744 + 3 19:48 mydate
-rw-rw-r-- 1 jth jth 866 + 3 19:47 mydate.c
-rwxrwxr-x 1 jth jth 9016 + 3 19:48 pipe_ls
-rw-rw-r-- 1 jth jth 738 + 3 19:39 pipe_ls.c
-rwxrwxr-x 1 jth jth 8864 + 3 19:48 printdir
-rw-rw-r-- 1 jth jth 279 + 3 19:48 printdir.c
```

## Part 2:

1. Edit the **builtin.c** file to recognize **cd**, **pwd**, **id**, **hostname** and **builtin**. Write functions implementing these commands, and compile then into your shell. The **builtin** command lists the functions built into your shell.

*Files provided:*

*builtin.c, parse.c, shell.c, shell.h*

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part2$ ./myshell
myshell -> cd /etc
myshell -> pwd
/etc
myshell -> id
UserID = 1000(jthong), GroupID = 1000(jthong)
myshell -> hostname
hostname: jth-VirtualBox
myshell -> builtin quit
quit is a builtin feature.
myshell -> builtin pwd
pwd is a builtin feature.
myshell -> builtin abc
abc is NOT a builtin feature.
myshell -> quit
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part2$
```

2. Modify the **redirect\_in.c** file to recognize standard input and **redirect\_out.c** file to recognize standard output. Add code to the **pipe\_present.c** file to check for the pipe symbol. Add code to the **pipe\_command.c** file to create a process to execute each of the pipe. Modify **is\_background.c** to check the "&" symbol. Alter the **run\_command.c** file to call these functions.

*Files provided:*

*redirect\_in.c, redirect\_out.c, run\_command.c, pipe\_present.c, pipe\_command.c, is\_background.c*

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part2$ ./myshell
myshell -> cat 123 > mess
myshell -> cat mess | sort -u
myshell -> APPLE
BUG
CANDY
DEFINE
ENTER
FINISH

myshell -> cat mess
CANDY
ENTER
APPLE
DEFINE
BUG
FINISH
myshell -> wc -l < mess
6
myshell -> quit
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part2$
```

### Part3:

1. The first look up project lab familiarizes you with the format of the dictionary by completing the **convert.c** program that creates the dictionary of fixed-length records (fixrec) from a file of variable-length entries (dict). Add code to **convert.c** to change an editable file into a fixed-length record format.

*File provided:*

*convert.c, dict.h, dict*

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part3$ ./convert dict myfixrec
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part3$ ls -l dict myfixrec
-rw-rw-r-- 1 jth jth 1920 + 5 2015 dict
-rw-rw-r-- 1 jth jth 10240 + 3 20:14 myfixrec
```

2. Add code to the **lookup1.c** file to do a simple linear search through a file of fixed length records. Link with **main.c**, the user interface module.

*File provided:*

*lookup1.c, main.c*

Sample output:

```
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part3$ ./file_lookup myfixrec
What word do you want : work
work : The curse of the drinking classes.

What word do you want : cynic
cynic : A blackguard who sees things as they are and not as they ought to be.

What word do you want : beauty
beauty : The power by which a woman charms a lover and terrifies a husband.

What word do you want : homework
homework : Not Found!

What word do you want : ^C
jth@jth-VirtualBox:~/Desktop/M043040098_SP_HW2/part3$
```

### Rules:

1. Please use **C** language in this homework and run your program on **Ubuntu 16.04**.
2. Please provide Makefile to compile your homework; otherwise, you will get **ZERO**.
3. **Do not copy others homework.**
4. If you have any question, please send email to [sp\\_ta@net.nsysu.edu.tw](mailto:sp_ta@net.nsysu.edu.tw) or come to F5018, but TA does not help to debug.

### Upload:

1. Please compress your homework into **zip** or **tar** archive.
2. Naming rules: "**StudentID\_SP\_HW2.zip**". For example: M033040001\_SP\_HW2.zip
3. Upload your homework to **NSYSU Cyber University**.
4. **Deadline: 2016/10/11 (Tue.) 23:59; if it is overdue, you will also get ZERO.**