

Name : Ngoun Mengleang

Class : ITE-Y3-M

Report OS lab 003

1. Write a shell script program to display a list of user currently logged in.

```
1  #!/bin/bash
2  echo $USER #variable USER is stored current user name
3  exit #exit use to terminate shell
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ± ./task1.sh
leang
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ±
```

2. Compare the running time of c compile file and shell script the programs using time command. Running time C compile file is faster than shell script.

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3/task2 main ± time ./TASK2A
Hello
./TASK2A 0.00s user 0.01s system 91% cpu 0.006 total
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3/task2 main ± time ./task2b.sh
Hello
./task2b.sh 0.01s user 0.00s system 92% cpu 0.014 total
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3/task2 main ±
```

3.
 - a. Display a type of file.

```
#!/bin/bash
2  #./task3b.sh [filename1 filename2 ..... ]
3  for n in "$@" # "$@" : "$1" "$2" "$3" .....
4  do
5  file "$n" #file is use to determine file type
6  done
7  exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ± ./task3a.sh task10.sh task1.sh OS_Lab3\ -\ Bash\ Programming.pdf Text\ A.txt
task10.sh: Bourne-Again shell script, ASCII text executable
task1.sh: Bourne-Again shell script, ASCII text executable
OS_Lab3 - Bash Programming.pdf: cannot open 'OS_Lab3 - Bash Programming.pdf' (No such file or directory)
Text A.txt: ASCII text
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ±
```

Name : Ngoun Mingleang

Class : ITE-Y3-M

b. Convert filename to uppercase.

```
1 #!/bin/bash
2 #./task3b.sh [filename1 filename2 ..... ]
3 for n in "$@" # take element in set "$@" one by one assign to $n
4 do # start loop task
5 mv "${n}" "${n^^}" # ${n^^} convert a whold string to uppercase
6 done # end loop task
7 exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ] ls
'os_lab3 - bash programming.pdf' task1.sh task3a.sh task3c.sh task4b.sh task6.sh task8.sh 'Text A.txt'
task10.sh task2.sh task3b.sh task4a.sh task5.sh task7.sh task9.sh TextB.txt
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ] ./task3b.sh os_lab3\ -\ bash\ programming.pdf task1.sh
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ] ls
'OS_LAB3 - BASH PROGRAMMING.PDF' TASK1.SH task3a.sh task3c.sh task4b.sh task6.sh task8.sh 'Text A.txt'
task10.sh task2.sh task3b.sh task4a.sh task5.sh task7.sh task9.sh TextB.txt
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ]
```

c. shell script that determines the duration for which a specified user is working on the system.

```
1 #!/bin/bash
2 #./task3c.sh [Username]
3 userloginTime=$(last --time iso|grep -w "$1"|grep -w "still logged in"|awk '/tty/ {print $4}') #get time that user login
4 UnixTimeStart=$(date -d "$userloginTime" +%s") #covert userloginTime to Unix Time
5 UnixTimeNow=$(date +%s") #get Unix Time now
6 Period=$(expr $UnixTimeNow - $UnixTimeStart) #calculate duration from user login to now
7 day=$(expr $Period % 31556926 % 2629743 / 86400) #covert duration to day
8 hour=$(expr $Period % 31556926 % 2629743 % 86400 / 3600) #covert duration to hour
9 minu=$(expr $Period % 31556926 % 2629743 % 86400 % 3600 / 60) #covert duration to minu
10 sec=$(expr $Period % 31556926 % 2629743 % 86400 % 3600 % 60) #covert duration to sec
11 echo "User use this computer : " $day "Day" $hour "Hous" $minu "Minute" $sec "Second"
12 exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ] ./task3c.sh leang
User use this computer : 0 Day 0 Hous 49 Minute 24 Second
leang@leang-x550vx ~ /RUPP/OS_RUPP/lab/lab3 [ main ]
```

Name : Ngoun Mingleang

Class : ITE-Y3-M

4.

a. displays all the lines between the given line numbers.

- Sed -n is terminate auto printing pattern space
- 'p' is an explicit print. start,end\p print line between start to end.

```
1  #!/bin/bash
2  #./task4a.sh [filename] [startline] [endline]
3  fname=$1
4  start=$2
5  end=$3
6  sed -n $start,$end\p "$fname"      #sed -n p textfile use to print data that store in patter space
7  exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ± cat TextB.txt
Alice everyday goes to library to study
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
They have lost thousands of jobs to Asia, South America, and Mexico
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ± ./task4a.sh TextB.txt 2 4
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
They have lost thousands of jobs to Asia, South America, and Mexico
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ±
```

b. deletes all lines containing a specified word in one or more files supplied as arguments to it.

- sed -i : edit file
- d : delete pattern space. 1 line is 1 pattern space

```
1  #!/bin/bash
2  #./task4b.sh [world] [filename1 filename2 .....]
3  world=$1
4  for n in "${@:2}"      #start form 2nd arguments
5  do
6  sed -i "/$1/d" "$n"     #sed -i edit file and d is delete pattern space
7  done
8  exit
9
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ± cat Text\ A.txt
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
A result (also called upshot) is the final consequence of a sequence of
actions or events expressed qualitatively or quantitatively.
Possible results include advantage, disadvantage, gain, injury, loss, value and victory.
There may be a range of possible outcomes associated with an event depending on the point of view,
historical distance or relevance. Reaching no result can mean that actions are inefficient, ineffective, meaningless or flawed.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ± ./task4b.sh actions Text\ A.txt
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ± cat Text\ A.txt
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
A result (also called upshot) is the final consequence of a sequence of
Possible results include advantage, disadvantage, gain, injury, loss, value and victory.
There may be a range of possible outcomes associated with an event depending on the point of view,
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 ~ main ±
```

Name : Ngoun Mengleang

Class : ITE-Y3-M

5. Write a shell Script program to check whether the given number is even or odd.
 - a. `expr` : evaluate expressions
 - b. `-eq` : equal operator

```
1  #!/bin/bash
2  #./task5.sh [number]
3  number=`expr $1 % 2`      #use expr for calculate with two argument
4  if [ $number -eq 0 ]      # -eq is ==
5  then
6      echo $1 " is even"
7  else
8      echo $1 " is odd"
9  fi
10 exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main] ./task5.sh 8
8 is even
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main] ./task5.sh 7
7 is odd
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main]
```

6. Write a shell script Program to search whether element is present in the list or not.
 - `grep -w` : select one line that contain a whole word
 - `$?` : check return value of last command that executed. In this case `$? == 0` : success and `$? == 1` failure

```
1  #!/bin/bash
2  #./task6.sh [word] [filename]
3  grep -w "$1" "$2" #grep -w get one line that contain $1
4  if [ $? == "0" ]
5  then
6      echo $1 " is in this list"
7  else
8      echo $1 " isn't in this list"
9  fi
10 exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main] cat TextA.txt
Perhaps
the
decline
of
this
country
has
already
started

leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main] ./task6.sh of TextA.txt
of is in this list
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main] ./task6.sh dog TextA.txt
dog isn't in this list
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [main]
```


Name : Ngoun Mengleang

Class : ITE-Y3-M

7. Check two have same content or not

a. If same delete 2nd file

- diff : compare file line by line
- command > /dev/null: redirects the output of command(stdout) to /dev/null
- 2>&1: redirects standard error to standard output, so errors (if any) also goes to /dev/null

```
1 #!/bin/bash
2 #./task7.sh [file1] [file2]
3 diff "$1" "$2" > /dev/null 2>&1 #lock result
4 if [ $? -eq "0" ] # $? is a exit status of the last command executed. $? == 0 mean dosen't have status. $? == 1 mean have status
5 then
6     echo $1 " and " $2 " are a same file."
7     rm "$2"
8 elif [ $? -eq "1" ]
9 then
10    echo $1 " and " $2 " aren't a same file."
11    cat "$1" >> "$2"
12 fi
13 exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± cat TextA.txt
In mathematics and statistics, the arithmetic mean,
or simply the mean or the average, is the sum of a collection of
numbers divided by the count of numbers in the collection.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± cat TextB.txt
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
They have lost thousands of jobs to Asia, South America, and Mexico.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± ./task7.sh TextA.txt TextB.txt
TextA.txt and TextB.txt aren't a same file.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± cat TextB.txt
Perhaps the decline of this country has already started
The management of your company has executed its duties excellently
They have lost thousands of jobs to Asia, South America, and Mexico.In mathematics and statistics, the arithmetic mean,
or simply the mean or the average, is the sum of a collection of
numbers divided by the count of numbers in the collection.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ±
```

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± cat TextA.txt
In mathematics and statistics, the arithmetic mean,
or simply the mean or the average, is the sum of a collection of
numbers divided by the count of numbers in the collection.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± cat TextB.txt
In mathematics and statistics, the arithmetic mean,
or simply the mean or the average, is the sum of a collection of
numbers divided by the count of numbers in the collection.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± ./task7.sh TextA.txt TextB.txt
TextA.txt and TextB.txt are a same file.
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ± ls
'OS_LAB3 - BASH PROGRAMMING.PDF' TASK1.SH task3a.sh task3c.sh task4b.sh task6.sh task8.sh TextA.txt
task10.sh task3b.sh task4a.sh task5.sh task7.sh task9.sh
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 [0] main ±
```

Name : Ngoun Mengleang

Class : ITE-Y3-M

8. Develop an interactive script that ask for a word and a file name and then tells how many times that word occurred in the file
 - `grep -o` : print only match word
 - `Wc -l` : count line

```
1  #!/bin/bash
2  #./task8.sh [world] [filename]
3  time=$(grep -wo "$1" "$2"|wc -l) #grep -wo is for select only match word and wc -l count amount line
4  echo "World" $1 "is occurred in " $2 " " $time " time"
5  exit
```

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ± ./task8.sh or TextA.txt
World or is occurred in TextA.txt 2 time
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ±
```

9. Write a shell script that operate string (string length , substring)

```
leang@leang-x550vx ~/RUPP/OS_RUPP/lab/lab3 main ± ./task9.sh
Task9
Input Sting : selection
1 : To extract a sub-string from a given string
2 : To find the length of a given string
Input menu : 1
1 : Extract a sub string from n to m
2 : Extract a sub string from n to last index
Input menu : 1
Input start index : 2
Input end index : 5
lec
1 : To extract a sub-string from a given string
2 : To find the length of a given string
Input menu : 1
1 : Extract a sub string from n to m
2 : Extract a sub string from n to last index
Input menu : 2
Input position : 3
ection
1 : To extract a sub-string from a given string
2 : To find the length of a given string
Input menu : 2
String's length : 9
1 : To extract a sub-string from a given string
2 : To find the length of a given string
Input menu : ^C
```

```
#!/bin/bash
# ${#variable} find length of variable
# ${string:n} split substring from n index
# ${string:n:length} split substring from n with length
echo "Task9"
echo -n "Input Sting : "
read string
while [ true ]
do
    echo "1 : To extract a sub-string from a given string"
```

Name : Ngoun Mingleang

Class : ITE-Y3-M

```
echo "2 : To find the length of a given string"
```

```
echo -n "Input menu : "
```

```
read menu
```

```
if [ $menu == 1 ]
```

```
then
```

```
    echo "1 : Extract a sub string from n to m"
```

```
    echo "2 : Extract a sub string from n to last index"
```

```
    echo -n "Input menu : "
```

```
    read subMenu
```

```
    if [ $subMenu == 1 ]
```

```
    then
```

```
        echo -n "Input start index : "
```

```
        read s_position
```

```
        echo -n "Input end index : "
```

```
        read e_position
```

```
        length=`expr $e_position - $s_position`
```

```
        if [ $length -gt ${#string} ]
```

```
        then
```

```
            echo "position out of bound"
```

```
        else
```

```
            echo ${string:s_position:length}
```

```
        fi
```

```
    elif [ $subMenu == 2 ]
```

```
    then
```

```
        echo -n "Input position : "
```

```
        read position
```

```
        if [ $position -gt ${#string} ]
```

```
        then
```

```
            echo "position out of bound"
```

```
        else
```

```
            echo ${string:position}
```

```
        fi
```

```
    fi
```

```
elif [ $menu == 2 ]
```

```
then
```

```
    echo "String's length : " ${#string}
```

```
fi
```

```
done
```

```
exit
```

Name : Ngoun Mengleang

Class : ITE-Y3-M

10. Write a shell script program to display the process attributes

- `ps -fu` : report a snapshot of the current processes of specific user in full form list

```
lab > lab3 > task10.sh
1  #!/bin/bash
2  #./task10.sh [username]
3  ps -fu "$1" # ps -fu use for display current processes in full format for user $1
4

leang@leang-x550vx: ~/RUPP/OS_RUPP/terminal$ ./task10.sh leang
UID          PID    PPID  C  STATE  TIME  CMD
leang        1125     1    0  20:44 ?    00:00:00 /usr/lib/systemd/systemd --user
leang        1127    1125    0  20:44 ?    00:00:00 (sd-pam)
leang        1137    1125    3  20:44 ?    00:02:50 /usr/bin/pulseaudio --daemonize=no --log-target=journal
leang        1142     1    0  20:44 ?    00:00:01 /usr/bin/kwalletd5 --pam-login 7 3
leang        1143    1123    0  20:44 ?    00:00:00 /usr/bin/startplasma-x11
leang        1149    1125    0  20:44 ?    00:00:02 /usr/bin/dbus-daemon --session --address=systemd: --nofork --nopidfile --systemd-activation --syslog-only
leang        1151    1125    0  20:44 ?    00:00:00 /usr/lib/gvfsd
leang        1156    1125    0  20:44 ?    00:00:00 /usr/lib/gvfsd-fuse /run/user/1000/gvfs -f
leang        1200     1    0  20:44 ?    00:00:00 /usr/lib/kf5/start_kdeinit
leang        1201     1    0  20:44 ?    00:00:00 kdeinit5: Running...
leang        1202    1201    0  20:44 ?    00:00:02 /usr/lib/kf5/klauncher --fd=9
leang        1221    1137    0  20:44 ?    00:00:00 /usr/lib/pulse/gsettings-helper
leang        1230     1    0  20:44 ?    00:00:05 /usr/bin/kded5
leang        1234     1    9  20:44 ?    00:08:23 /usr/bin/kwin_x11
leang        1249    1125    0  20:44 ?    00:00:01 /usr/lib/kactivitymanagerd
leang        1256     1    0  20:44 ?    00:00:02 /usr/bin/kmsserver
leang        1258    1125    0  20:44 ?    00:00:02 /usr/bin/kglobalaccel5
leang        1264     1    1  20:44 ?    00:01:35 /usr/bin/plasmashell
leang        1268     1    0  20:44 ?    00:00:02 /usr/bin/kaccess
leang        1270     1    0  20:44 ?    00:00:01 /usr/lib/polkit-kde-authentication-agent-1
leang        1272     1    0  20:44 ?    00:00:02 /usr/bin/baloo_file
leang        1276     1    0  20:44 ?    00:00:01 /usr/bin/xembedsniproxy
leang        1286     1    0  20:44 ?    00:00:02 /usr/lib/kdeconnectd
leang        1288     1    0  20:44 ?    00:00:01 /usr/bin/gmenubsmenuproxy
leang        1293    1125    0  20:44 ?    00:00:00 /usr/lib/dconf-service
leang        1299    1256    0  20:44 ?    00:00:02 /usr/bin/msm_kde_notifier -session 10108c214f11b000160359037300000014290004_1605534158_808456
leang        1335    1256    0  20:44 ?    00:00:11 /usr/bin/yakuake -session 10108c214f11b000160493753600000014070005_1605534158_808515
leang        1349    1256    0  20:44 ?    00:00:01 /usr/lib/xdg-desktop-portal-kde -session 10108c214f11b000160493753700000014070007_1605534158_808774
leang        1352    1125    0  20:44 ?    00:00:01 /usr/lib/kf5/kscreen_backend_launcher
leang        1353    1125    0  20:44 ?    00:00:00 /usr/lib/bluetooth/obexd
leang        1360    1256    0  20:44 ?    00:00:01 /usr/lib/baloorunner -session 10108c214f11b000160493815800000014070012_1605534158_808549
leang        1373     1    0  20:44 ?    00:00:00 /usr/bin/pamac-tray-appindicator
leang        1377     1    0  20:44 ?    00:00:11 /opt/xdman/jre/bin/java -Xmx1024m -jar /opt/xdman/xdman.jar -m
leang        1381     1    0  20:44 ?    00:00:02 /usr/lib/org_kde_powerdevil
leang        1393     1    0  20:44 ?    00:00:00 /usr/lib/geoclue-2.0/demos/agent
leang        1447    1125    0  20:44 ?    00:00:00 /usr/lib/at-spi-bus-launcher
leang        1453    1447    0  20:44 ?    00:00:00 /usr/bin/dbus-daemon --config-file=/usr/share/defaults/at-spi2/accessibility.conf --nofork --print-address 3
leang        1455    1335    0  20:44 pts/1 00:00:00 /usr/bin/zsh
leang        1457    1125    0  20:44 ?    00:00:01 /usr/lib/xdg-desktop-portal
leang        1459    1125    0  20:44 ?    00:00:02 /usr/lib/at-spi2-registrard --use-gnome-session
```


Name : Ngoun Mingleang

Class : ITE-Y3-M