Operating System Overview: Unix
EPITA
Homework as replacement for examination

You must upload all scripts and answers inside teams before the 16/05/2022 at 23h59.
If you copy the answer of someone else, both persons will get the same mark (0/20).

Exercise 1

Write a script bash that displays the following menu (in loop):

1) List of users in your ubuntu box
2) Kernel version
3) Distribution name and version
4) Identity of the user
5) Privilege escalation to administrator
6) Exit

Select your choice:**1**
nobody
oscar
user1
Select your choice:**2**
5.8.0-40-generic

Select your choice: **3**
Distributor ID: Ubuntu
Description: Ubuntu 20.10
Release: 20.10
Codename: groovy

The script must return the output of commands as displayed in the previous execution.

Exercise 2

2.1 Write a script bash which finds and removes duplicated files in the same directory. As the duplicated files are in the current directory, they do not have the same name but the same content.

2.2 Write a script that takes two operands and an operator (+, -, x, /, //, pow, mod) as argument or by prompting the user when no argument is provided and displays the result of the desired operation in the terminal.

2.3 Write a script that takes as input an alpha-numerical chain from user (e.g., 123abc456) and transforms lower case letters in the chain into upper case letters.

2.4 Write a script which takes as argument a number between [1,12] and return the number of days in the given month.

2.5 Write a script that takes as input a decimal number and converted into binary, octal and hexadecimal.

2.6 Write a script which takes a variable number of arguments and return them sorted in ascending order.

2.7 Write a bash script that displays the number of characters in each line of a file passed as an argument and displays an error if no argument is provided.

2.8 Write a bash script that takes a word as an argument, and check if this word exists in English. The English dictionary is in: /usr/share/dict/american-english

2.9 Write a bash script that prompts user for time and distance, and it derives the speed with 2 decimals.

2.10 Write a bash script that prompts user for his first name and birth year, and it returns his name & his age.

Exercise 3

Give the script and the displayed output of each of the following scripts:

1) ***$cat script1.sh***

```
#!/bin/bash
Person=swathy
```

```
echo $Person
echo '$Person'
echo "$Person"
echo \$Person
```

PS1 > ./script1.sh

2) ***$cat script2.sh***

```
#!/bin/bash
a=1
if [ $a -eq 1 ]; then echo "a is equal to $a"; fi
for i in {1..5}; do echo "Hello World"; done
while [ $i -le 10 ]; do echo $i; sleep 2; done
```

PS1 > ./script2.sh

3) $cat script3.sh

```
#!/bin/bash
while [ $# -gt 1 ]
do
 case $1 in
  -a) ip=$2
            shift 2
  ;;
  -p) port=$2
            shift 2
  ;;
  *) echo "unrecognized arguments"
  ;;
 esac
done
echo "Address:$ip and port: $port"
```

PS1 > ./script3.sh

4) $cat script4.sh

```
[ $# -le 1 ] && set -- student master epita
for i
do
  echo "Looping ... number $i from $#"
done
```

PS1 > ./script4.sh

5) $cat script5.sh

```
 #!/bin/bash
. script2.sh
while [ $i -le 15 ]
do
    echo $i
    ((i++))
done
```

PS1 > ./script5.sh

6) $cat script6.sh

```
#!/bin/bash
var="This is a text
Over several
lines"
echo var
echo $var
echo "$var"
echo '$var'
echo ${#var}
echo ${var:20:5}
```

```
var="2*(5+5) "
echo $var
echo "$var"
echo $(($var))
```

PS1 > ./script6.sh

7) $cat script7.sh

```
#!/bin/bash
for i in `seq 8`
do
   echo -n "$i …"
   t=$((2*var+20))
   echo $((t % 7 + 1 ))
done
echo DONE …
```

PS1 > ./script7.sh

8) $cat script8.sh

```
#!/bin/bash
for i in *.doc
do
        mv -v $i ${i%.doc}.txt
done
```

PS1 > touch f1.doc f2.doc f3.doc && ./script8.sh

9) $cat script9.sh

```
#!/bin/bash
n=$1
rev=0
sd=0
while [ $n -gt 0 ]
do
   sd=`expr $n % 10`
   rev=`expr $rev \* 10  + $sd`
   n=`expr $n / 10`
done
echo  "The resulted number is $rev"
```

PS1 > ./script9.sh 123

10) $cat script10.sh

```
#!/bin/bash
display ( ) {
echo 'First Block'
echo 'Number 1'
}
display ( ) {
echo 'Second Block'
echo 'Number 2'
}
display
exit 0
```

PS1 > ./script10.sh

Exercise 4

We consider the following array:

```
arr=(epita is the best school)
```

4.1 Give and explain the output of each of the following commands:

```
echo ${#arr}
echo ${#arr[*]}
echo ${arr}
```

```
echo ${arr[@]}
echo ${#arr[0]}
echo ${#arr[1]}
echo ${#arr[2]}
echo ${#arr[3]}
echo ${arr[@]:3}
echo ${arr[@]:2:2}
```

4.2 Give the command**s** to add the words ″first″, ″last″, ″middle″ to the beginning, end and middle of the array ″arr″.

4.3 Give the command**s** to remove the first and last elements from the array ″arr″.

4.4 Write a script bash to fill an empty array with five different random numbers between 1 and 50. The script generates a random number and checks if it is different from previously generated numbers inside the array before storing it or generating another one.

Exercise 5

5.1) Write a script Bash that takes as argument the following file and displays only the full name and the phone number.

```
$ cat file.txt
Pol AURELIEN !Admin !0612345678!45 rue jean jaures
Adrien RENARD !Technicien !0610111213!24 rue bateau
Marc LEVALLOIS! Manager !0620222426!30 rue de la liberte
…
```

5.2) Write a script Bash that takes as argument the following file and displays the name and the appreciation for each student in the following file (″marks.csv″). The appreciations are: A:excellent, B: good, C: Average, D: below average and F: Fail.

```
$head -7 marks.csv
Marwa;renard;17;14;12
Gwendal;lay;15;16;5
Sandrine;tureleti;18;17;19
Edward;robinet;7;8;5
Guillaume;sapin;13;3;14
Yacine;bellaiche;17;15;13
David;mitterand;12;5;9
…
```

Exercise 6

Give the output of the following commands (only the displaying result for commands that start with echo):

```
filename=Bash.bish.Bosh.boush.tar.gz
echo ${#filename }
echo ${filename:5}
echo ${filename:5:3}
echo ${#filename }
echo ${Filename:=lucas}
echo $Filename
echo ${FILENAME:-george}
echo $FILENAME
echo ${FILENAME:?File }
echo ${filename^}
echo ${filename^^}
echo ${filename,}
echo ${filename,,}
echo ${filename~}
echo ${filename~~}
echo ${filename#*sh.}
echo ${filename##*sh.}
echo ${filename%.b*}
echo ${filename%%.b*}
```

```
echo ${filename/sh/shell}
echo ${filename//sh/shell}
echo ${filename/sh }
echo ${filename//sh/shell}
echo ${filename:=10}
echo $filename
```