Bash Scripting Day 3 Assignment

Mohamed Emary

January 29, 2025

1 Assignment Questions

1.1 Question 1

> Write a script called `mycase`, using the case utility to checks the type of character entered by a user:
>
> a. Upper Case
> b. Lower Case
> c. Number
> d. Nothing

```bash
#!/usr/bin/bash

shopt -s extglob

echo "enter a char"
read ch

case $ch in
[a-z])
  echo "lowercase"
  ;;
[A-Z])
  echo "uppercase"
  ;;
[0-9])
  echo "number"
  ;;
*)
  echo "Nothing"
  ;;
esac
```

1.2 Question 2

> Enhanced the previous script, by checking the type of string entered by a user:
>
> a. Upper Cases
> b. Lower Cases
> c. Numbers
> d. Mix
> e. Nothing

```
#!/bin/bash

shopt -s extglob

echo Please type your string:
read input

case $input in
+([a-z]))
  echo Lower Case
  ;;
+([A-Z]))
  echo Upper Case
  ;;
+([0-9]))
  echo Number
  ;;
+([0-9]|[a-z]|[A-Z]))
  echo Mix
  ;;
*)
  echo Nothing
  ;;
esac
```

1.3 Question 3

> Write a script called `mychmod` using `for` utility to give execute permission to all files and directories in your home directory.

```
#!/bin/bash

echo "Press Enter to give execute permission to all files and directories
↪  in your home directory"
echo "Or press Ctrl+C to cancel"
read -r

for item in ~/*; do
  echo "$item"
  chmod +x "$item"
done
```

```
echo "Finished"
```

1.4 Question 4

> Write a script called `mybackup` using `for` utility to create a backup of only files in your home directory.

```
#!/bin/bash

echo "Press enter to start backup, or Ctrl+C to cancel"
read input
echo Backup started...

backup_folder="$HOME/home_backup"

mkdir "$backup_folder"

for item in "$HOME"/*; do
  if [ -f "$item" ]; then
    cp "$item" "$backup_folder"
  fi
done

tar -cvf "$HOME/$backup_folder.tar" "$HOME/home_backup"
rm -r "$backup_folder"
echo "Backup completed: $backup_folder.tar"
```

1.5 Question 5

> Write a script called `mymail` using for utility to send a mail to all users in the system.
> Note: write the mail body in a file called `mtemplate`.

```
#!/usr/bin/bash

users=$(awk -F: '{print $1}' /etc/passwd)
for user in $users; do
  echo "Sending mail to $user"
  cat "mtemplate" | mail -s "Test Mail" "$user"
done

echo "Finished"
```

1.6 Question 6

> Write a script called `chkmail` to check for new mails every 10 seconds.
> Note: mails are saved in `/var/mail/username`.

Since all emails of a user are stored in a single file this script stores the current checksum of the mail file and compares it with the previous checksum. If the checksums are different, it means that the mail file has been updated (the user has new mail).

```bash
#!/bin/bash

mail_file="/var/mail/$USER"
old_hash=""

while true; do
  if [ -f "$mail_file" ]; then
    current_hash=$(md5sum "$mail_file" | cut -d' ' -f1)

    if [ -n "$old_hash" ] && [ "$current_hash" != "$old_hash" ]; then
      echo "You have new mail!"
    fi

    old_hash=$current_hash
  else
    echo "Mail file does not exist"
  fi
  sleep 10
done
```

1.7 Question 7

> What is the output of the following script
>
> ```ksh
> #!/bin/ksh
>
> typeset -i n1
> typeset -i n2
>
> n1=1
> n2=1
>
> while test $n1 -eg $n2; do
>   n2=$n2+1
>   print $n1
>   if [ $n1 -gt $n2]; then
>     break
>   else
>     continue
>   fi
>   n1=$n1+1
>   print $n2
> done
> ```

The script above will print `1` once then the loop condition will be false and the loop will exit.

1.8 Question 8

> Create the following menu:
>
> a. Press `1` to `ls`
>
> b. Press `2` to `ls -a`
>
> c. Press `3` to `exit`
>
> Using `select` utility then `while` utility.

```
#!/bin/bash

select option in "ls" "ls -a" "exit"; do
  case $option in
  "ls")
    ls
    ;;
  "ls -a")
    ls -a
    ;;
  "exit")
    break
    ;;
  *)
    echo Wrong option
    ;;
  esac
done
```

1.9 Question 9

> Write a script called `myarr` that ask a user how many elements he wants to enter in an array, fill the array and then print it.

```
#! /bin/bash

echo "How many elements do you want to enter?"
read num

declare -a myarr
for ((i = 0; i < $num; i++)); do
  echo "Array element:"
  read user_input
  $myarr[$i] = $user_input
done
```

1.10 Question 10

> Write a script called `myavg` that calculate average of all numbers entered by a user.
> Note: use arrays

```
#!/bin/bash

```

```
echo "Enter the number of numbers:"
read num

declare -a myarr
sum=0

for ((i = 0; i < $num; i++)); do
  echo "enter the array element"
  read user_input

  myarr[$i]=$user_input
  sum=$((sum + user_input))
done

avg=$(echo "scale=2; $sum / $num" | bc)
echo "average is $avg"
```

1.11 Question 11

> Write a function called `mysq` that calculate square if its argument.

```
#!/bin/bash

function square {
  typeset -i sq
  ((sq = $1 * $1))
  echo $sq
}
echo "The square is: $(square $1)"
```