ASSIGNMENT 7.1

Write a shell script to print last twenty commands issued by the user. The user name is supplied as a command line argument to the script (use bash-history file)

```
#!/bin/bash
 username = $1
 loc = "/home/$username/.bash_history"
tail -n 20 $loc
```

Write a shell program which displays the message "welcome" and prints the date when you login to your system.

```
$ vi ~/.bashrc

echo "Welcome and today's date is $(date)"
```

Accept a string from the terminal and echo a suitable message if it doesn't have atleast 10 characters.

Ans)
```
#!/bin/bash
 read -p "Enter string : " str
 if [ 'expr "$ str" : '.*"' -lt 10 ]
 then
     echo " The string is short "

 else
     echo " The string is atleast 10 characters long"

 fi
```

4. Write a shell script which gets executed the moment a user logs in. It should display the message "GOOD MORNING" or "GOOD AFTERNOON" or "GOOD EVENING" depending upon the time at which the user logs in.

Ans)
```
#!/bin/bash
  hour = $ (date + %H)
 if [ $ hour - lt 12 ]
 then
      echo " GOOD MORNING "
elif [ $ hour - lt 16 ]
then   echo " GOOD AFTERNOON "
elif [ $ hour - lt 20 ]
then   echo " GOOD EVENING "
fi
        $ vi ~/. bashrc
```

Week 8

```
#!/bin/bash
 dir = $1
 if [! -d "$dir"]
 then
       echo " Directory not found "
       exit 1
fi
files = $(find "$dir" - type f - size +100c - printf '%s %p\n'|
                                                  sort -rn)
 echo "$files"
 num = $(echo "$files" | wc - l)
 echo "Total number of files exceeding 100 bytes : $num"
 #!/bin/bash
  sum = 0
 for file in *
 do
     if [-f "$file"]
     then
          echo "$file"
          size = $(du -b "$file" |cut -f1)
          ((sum + = size))
   fi
 done
echo "Total size: $ sum bytes"
```

3.
```
echo "Required files are :"
ls | grep ^[aeiou]
```

4.
```
#!/bin/bash
if test $# -ne 2
then
      echo "Please give two filenames."
      exit
fi
if cmp -s "$1" "$2"
then
      echo "$1 and $2 are same"
      rm "$2"
else
      echo "$1 and $2 are not same"
fi
```

Week 9

```
#!/bin/bash

if [ $# -eq 0]
then
    echo "Please give filenames"
    exit
fi
λ = $1
Shift
for var1 in $*
do
   for var2 in `cat $1`
  do
      echo " Occurrence of $var2 in $var1 is: `grep -c $var2 $1`"
  done
done
```

```
#!/bin/bash
if [ $# -eq 0]
then
    echo "Please give filenames"
    exit
fi
p= "UNIX"
for file in $*
do
     sed "/$p/d" $file | tee ff
     mv ff $file
done
```

ASSIGNMENT 9.1

```
#!/bin/bash
if [ $# -eq 0]
then
    echo "Please provide a list of login names"
    exit
fi
while [ $# -gt 0 ]
do
   login_name = $1
   shift
  echo "Information for login name $login_name :".
  grep -w "$login_name" /etc/passwd
done
```

```
#!/bin/bash
read -p "Enter text :" text
word_count = $(echo $text | wc -w)
echo "No. of words of different lengths : $word_count"
```

4.

```
#! /bin /bash
if [ $# -eq 0 ]
then
    echo "Please give filename"
   exit
fi
echo -e "Word \t count"
cat $1 | tr -s ' ' '\n' | sort | uniq -c | sort -r |
awk '{ print $2 "\t" $1}'
```

5.

```
#! /bin /bash
if [ $ (( $# % 2)) -ne 0 ]
then
    echo "Error : odd no. of filenames supplied."
    exit 1
fi
for ((i=1; i<= $# ; i+=2));
do
   j=$((i+1))
  cp "${!i}" "${!j}"
done
```

ASSIGNMENT 10.1

1.
```
#!/bin/bash
PASSWORD = "defaultpass"
CHANCES = 3

echo -e "\033[1m"
echo "1. Number of users currently logged in"
echo "2. Calendar of current month"
echo "3. Date in the format : dd/mm/yyyy"
echo "4. Quit"
for ((i=1; i<= $CHANCES ; i++))
do
    read -sp "Enter password :" pass
    echo ""

    if ["$pass" = "$PASSWORD"]
    then
        echo "Password is correct."
        read -p "Enter choice : " ch
        case $ch in
        1) echo "Number of users currently logged in :
                $(who | wc -l)";;
      2) cal ;
    3) echo "Date in the format : $(date +%d/%m/%Y)";;
    4) exit ;;
    *) echo " Invalid choice";;
  esac
 exit;
  else
    echo " Incorrect password. You have $(($CHANCES - $i))
              attempts left."
  fi
 done
 echo "Maximum number of chances exceeded. Goodbye"
 exit
```