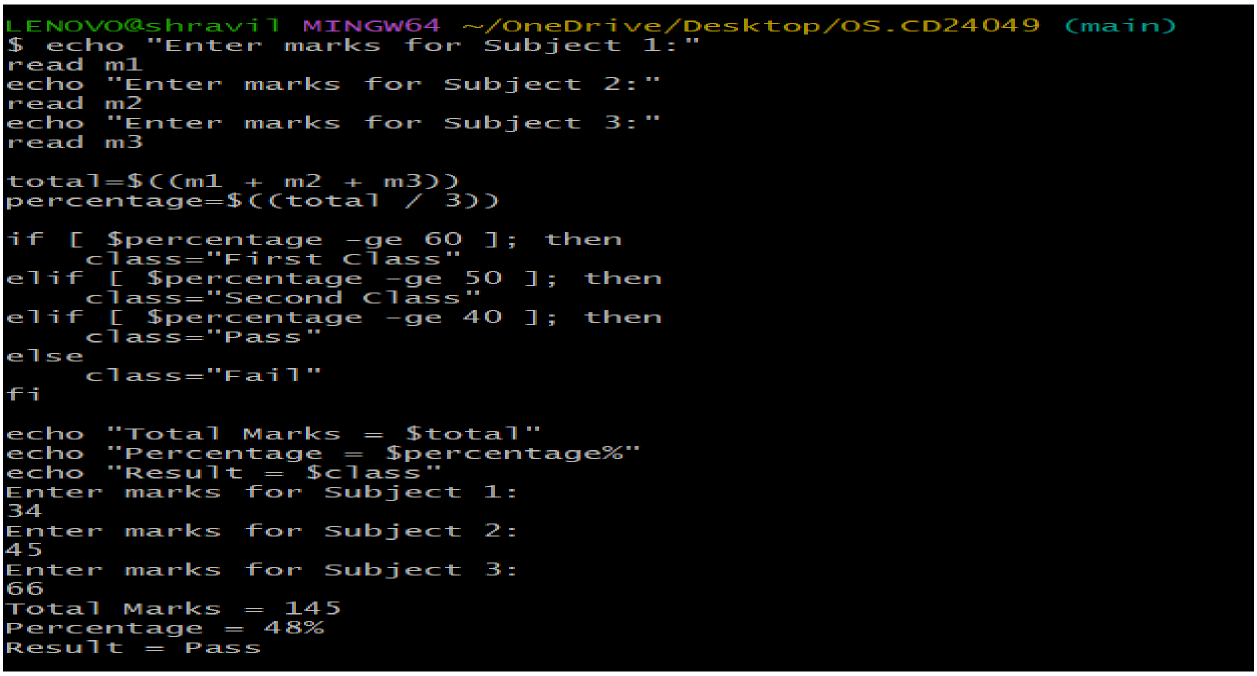
a] Write a shell script to generate mark-sheet of a student. Take 3 subjects, calculate and display total marks, percentage and Class obtained by the student.

Code:

```
echo "Enter marks for Subject 1:"

read m1

echo "Enter marks for Subject 2:"

read m2

echo "Enter marks for Subject 3:"

read m3

total=$((m1 + m2 + m3))

percentage=$((total / 3))

if [ $percentage -ge  60 ]; then

    class="First Class"

elif [ $percentage -ge  50 ]; then

    class="Second Class"

elif [ $percentage -ge  40 ]; then

    class="Pass"

else

    class="Fail"

fi
```

OUTPUT:



```
LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ echo "Enter marks for Subject 1:"
read m1
echo "Enter marks for Subject 2:"
read m2
echo "Enter marks for Subject 3:"
read m3

total=$((m1 + m2 + m3))
percentage=$((total / 3))

if [ $percentage -ge 60 ]; then
    class="First Class"
elif [ $percentage -ge 50 ]; then
    class="Second Class"
elif [ $percentage -ge 40 ]; then
    class="Pass"
else
    class="Fail"
fi

echo "Total Marks = $total"
echo "Percentage = $percentage%"
echo "Result = $class"
Enter marks for Subject 1:
34
Enter marks for Subject 2:
45
Enter marks for Subject 3:
66
Total Marks = 145
Percentage = 48%
Result = Pass
```

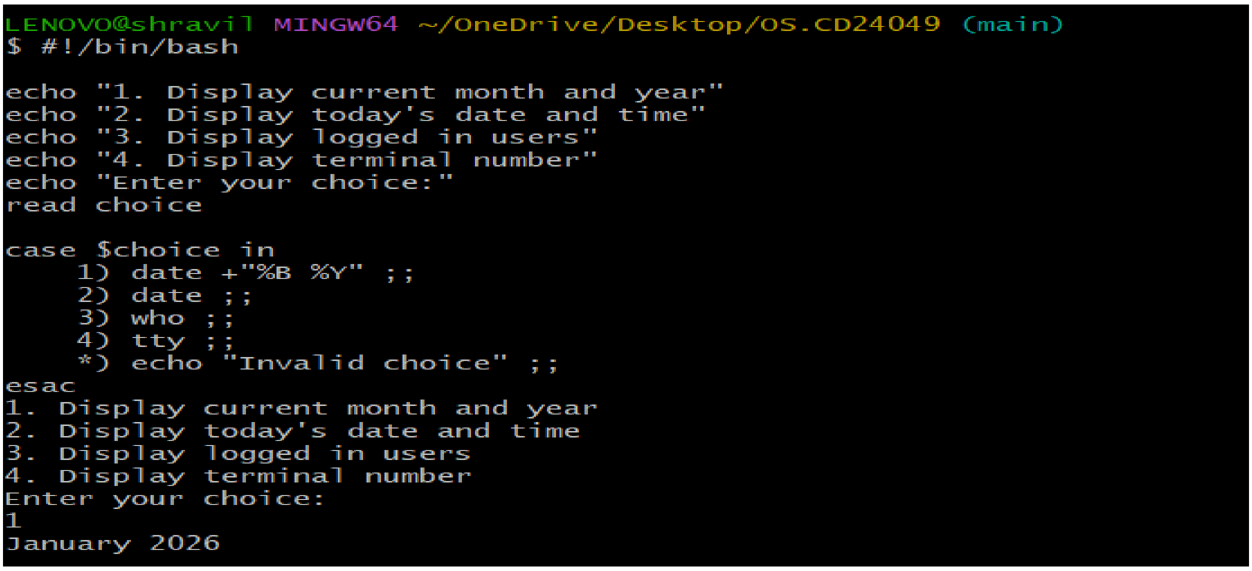
b] Write a menu driven shell script which will print the following menu and execute the given task.

- Display calendar of current month.
- Display today’s date and time.
- Display usernames those are currently logged in the system.
- Display your terminal number.

CODE:

```
echo "1. Display current month calendar"

echo "2. Display today's date and time"

echo "3. Display logged in users"

echo "4. Display terminal number"

echo "Enter your choice:"

read choice

case $choice in

1) cal ;;

2) date ;;

3) who ;;

4) tty ;;

*) echo "Invalid choice" ;;

Esac
```

OUTPUT:



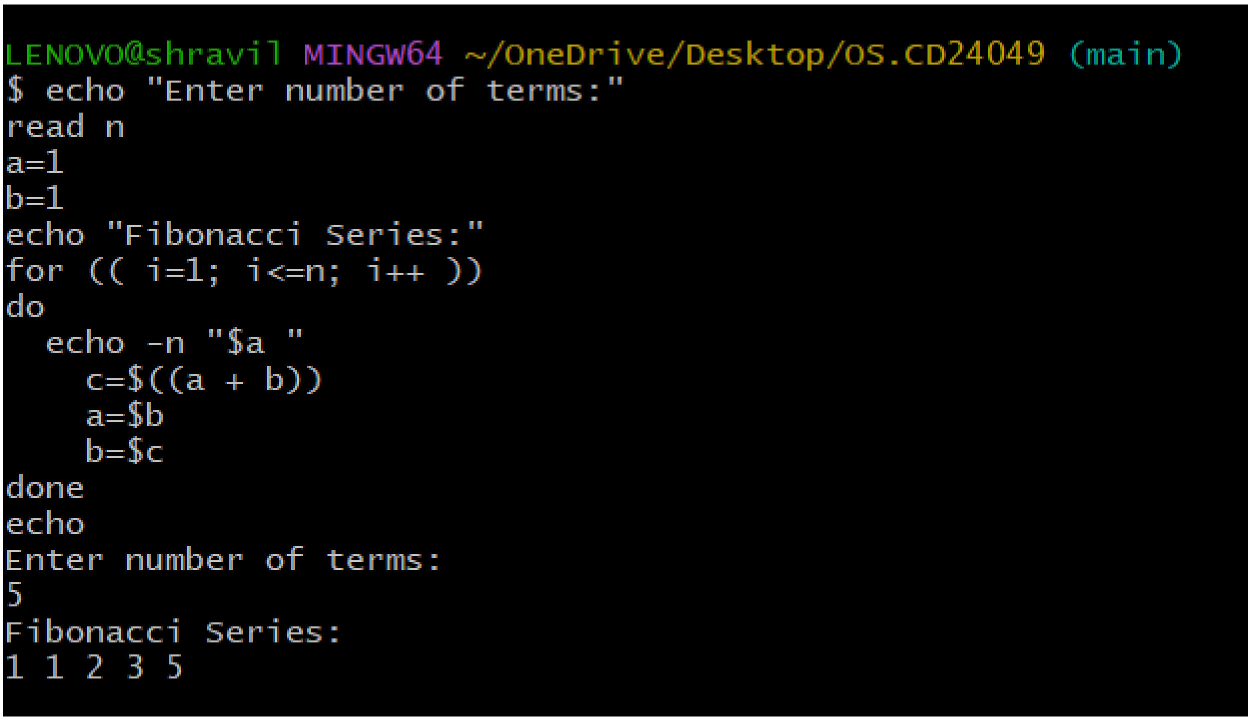
```
LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ #!/bin/bash

echo "1. Display current month and year"
echo "2. Display today's date and time"
echo "3. Display logged in users"
echo "4. Display terminal number"
echo "Enter your choice:"
read choice

case $choice in
    1) date +"%B %Y" ;;
    2) date ;;
    3) who ;;
    4) tty ;;
    *) echo "Invalid choice" ;;
esac
1. Display current month and year
2. Display today's date and time
3. Display logged in users
4. Display terminal number
Enter your choice:
1
January 2026
```

C] Write a shell script which will generate first *n* Fibonacci numbers like: 1, 1, 2, 3, 5, 13

CODE:

```
echo "Enter number of terms:"

read n

a=1

b=1

echo "Fibonacci Series:"

for (( i=1; i<=n; i++ ))

do

  echo -n "$a "

    c=$((a + b))

    a=$b

    b=$c

done

echo
```

OUTPUT:



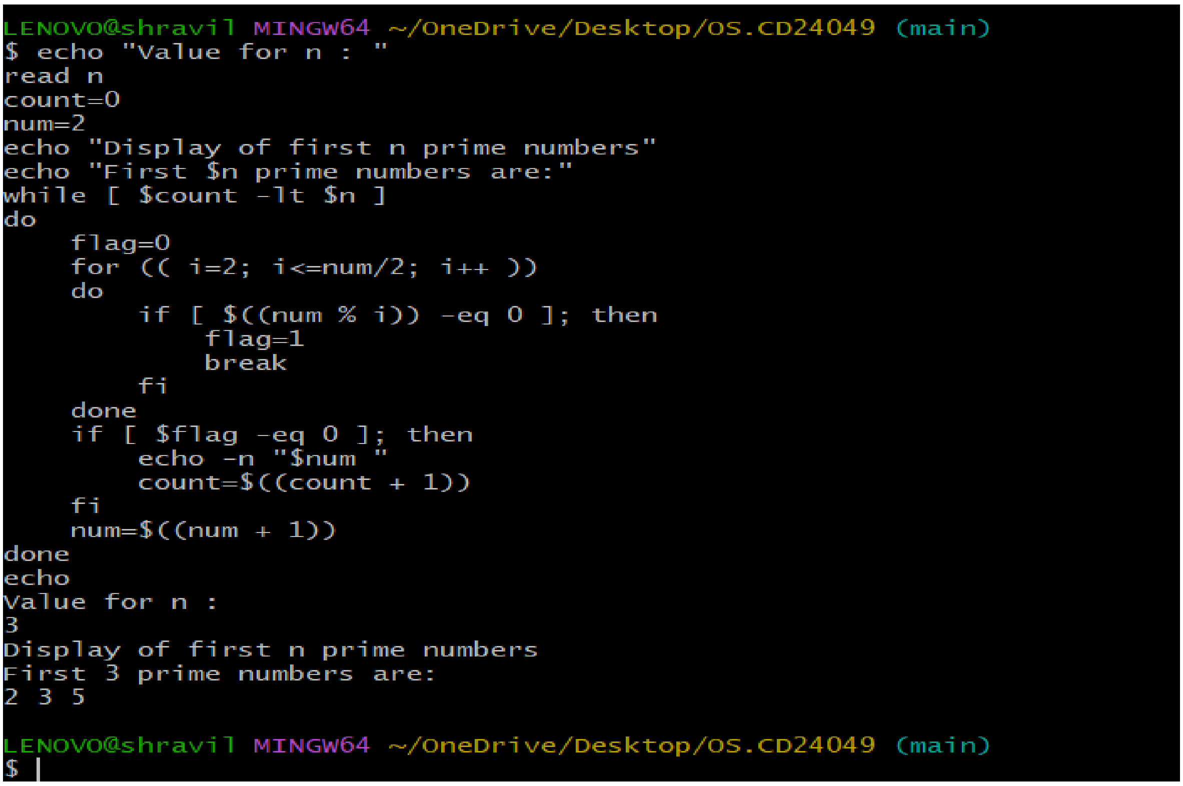
```
LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ echo "Enter number of terms:"
read n
a=1
b=1
echo "Fibonacci Series:"
for (( i=1; i<=n; i++ ))
do
  echo -n "$a "
    c=$((a + b))
    a=$b
    b=$c
done
echo
Enter number of terms:
5
Fibonacci Series:
1 1 2 3 5
```

D] Write a shell script which will accept a number *n* and display first *n* prime numbers as output.

CODE:

```
echo "Value for n : "
read n
count=0
num=2
echo "Display of first n prime numbers"
echo "First $n prime numbers are:"
while [ $count -lt $n ]
do
  flag=0
  for (( i=2; i<=num/2; i++ ))
  do
    if [ $((num % i)) -eq 0 ]; then
      flag=1
      break
    fi
  done
  if [ $flag -eq 0 ]; then
    echo -n "$num "
    count=$((count + 1))
  fi
  num=$((num + 1))
done
echo
```

OUTPUT:



```
LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ echo "Value for n : "
read n
count=0
num=2
echo "Display of first n prime numbers"
echo "First $n prime numbers are:"
while [ $count -lt $n ]
do
    flag=0
    for (( i=2; i<=num/2; i++ ))
    do
        if [ $((num % i)) -eq 0 ]; then
            flag=1
            break
        fi
    done
    if [ $flag -eq 0 ]; then
        echo -n "$num "
        count=$((count + 1))
    fi
    num=$((num + 1))
done
echo
Value for n :
3
Display of first n prime numbers
First 3 prime numbers are:
2 3 5

LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ |
```

e) Write menu driven program for file handling activity

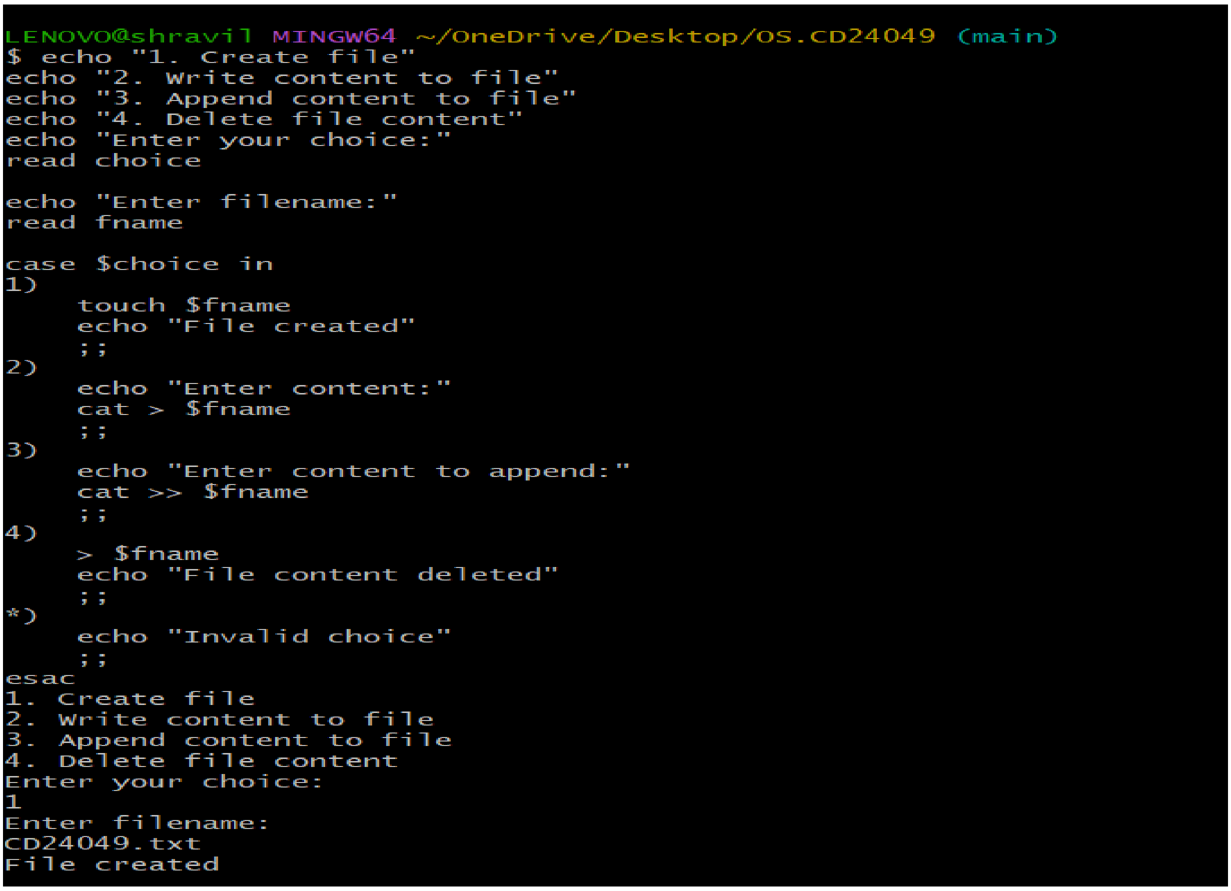
- Creation of file.
- Write content in the file.
- Upend file content.
- Delete file content.

CODE:

```
echo "1. Create file"

echo "2. Write content to file"

echo "3. Append content to file"

echo "4. Delete file content"

echo "Enter your choice:"

read choice

echo "Enter filename:"

read fname

case $choice in

1)
```

```
   touch $fname
   echo "File created" ;;
2)
   echo "Enter content:"
   cat > $fname ;;
3)
   echo "Enter content to append:"
   cat >> $fname;;
4)
   > $fname
   echo "File content deleted" ;;
*)
   echo "Invalid choice" ;;
esac
```

OUTPUT:



```
LENOVO@shravil MINGW64 ~/OneDrive/Desktop/OS.CD24049 (main)
$ echo "1. Create file"
echo "2. Write content to file"
echo "3. Append content to file"
echo "4. Delete file content"
echo "Enter your choice:"
read choice

echo "Enter filename:"
read fname

case $choice in
1)
    touch $fname
    echo "File created"
    ;;
2)
    echo "Enter content:"
    cat > $fname
    ;;
3)
    echo "Enter content to append:"
    cat >> $fname
    ;;
4)
    > $fname
    echo "File content deleted"
    ;;
*)
    echo "Invalid choice"
    ;;
esac
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
Enter your choice:
1
Enter filename:
CD24049.txt
File created
```