OS Practical No:03

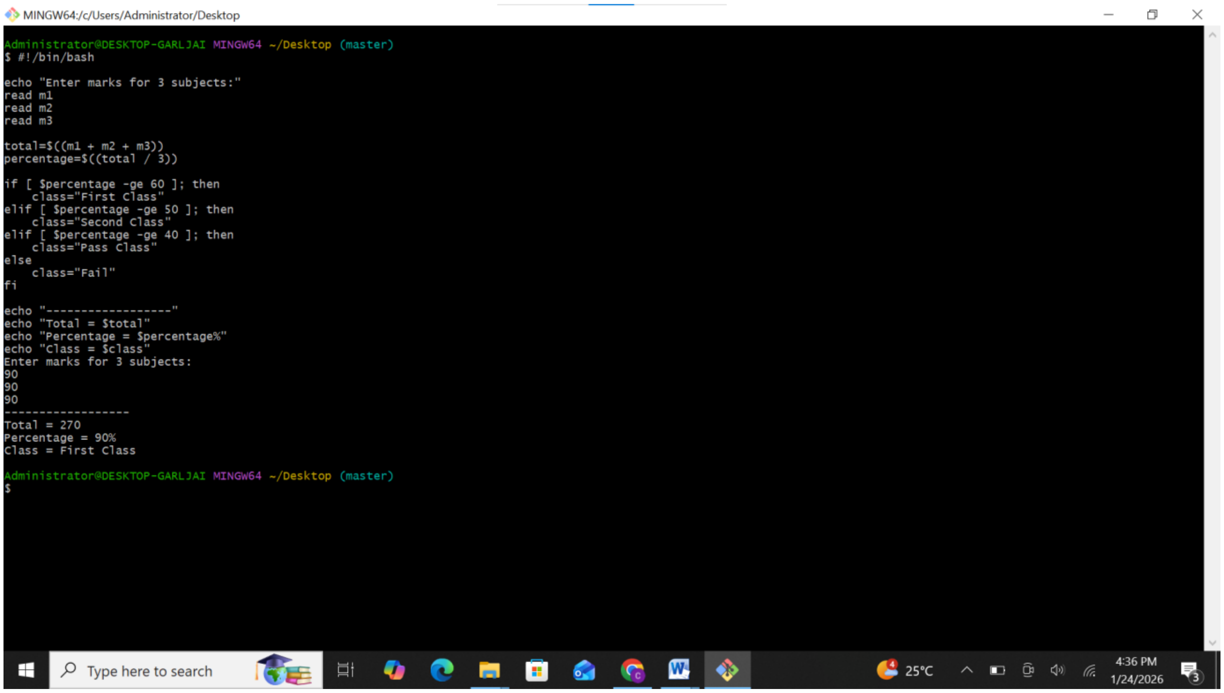
(Shell Scripting ased practical)

1.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 of English”
 read m1
echo “Enter marks of Maths”
read m2
echo “Enter marks of science” read
m3 total=$(($m1+$m2+$m3))
percentage=$((total/3))  echo
“Student: Total Marks =

$total” echo “Percentage =
$percentage”
if [ $percentage -gt 75 ]; then
          echo “Class: Distinction”
elif [ $percentage -gt 60 ]; then
           echo “Class: First Class”
elif [ $percentage -gt 40 ]; then
           echo “Class: Second Class”
elif [ $percentage -gt 35 ]; then
          echo “Class: Third Class”
else
      echo “Class: Fail”
     fi
```

Output:



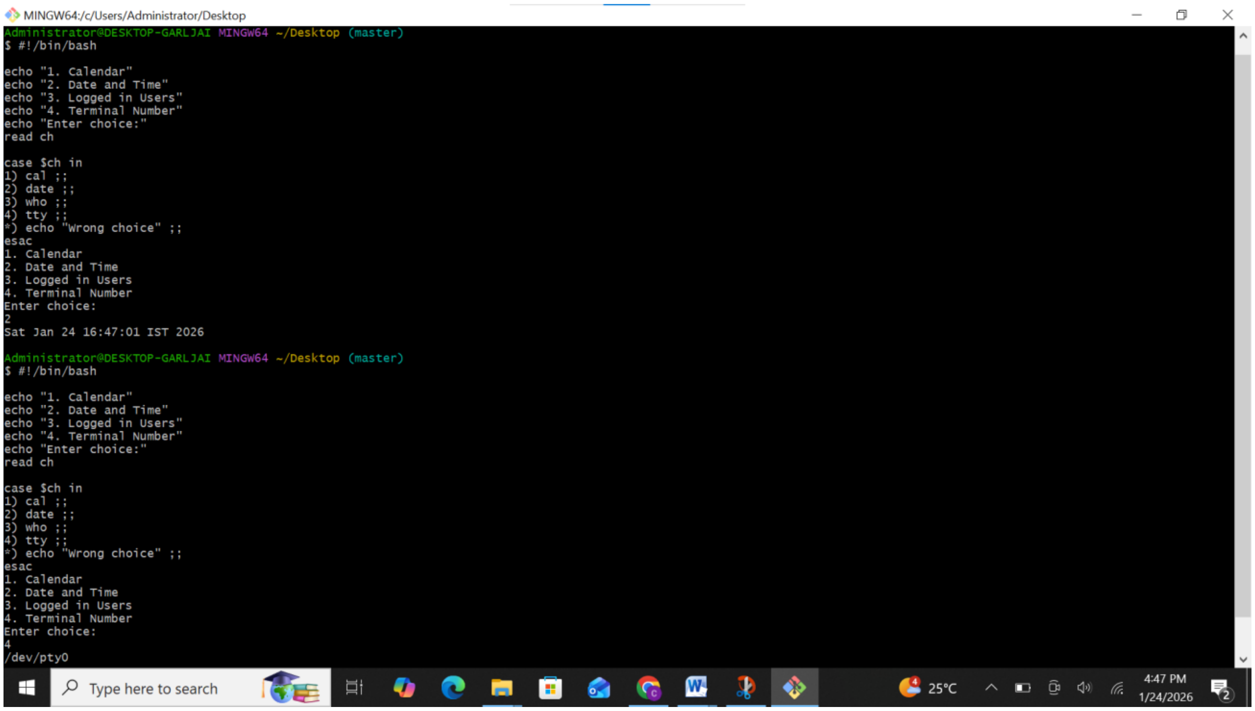
```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "Enter marks for 3 subjects:"
read m1
read m2
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 Class"
else
    class="Fail"
fi

echo "------------------"
echo "Total = $total"
echo "Percentage = $percentage%"
echo "Class = $class"
Enter marks for 3 subjects:
90
90
90
------------------
Total = 270
Percentage = 90%
Class = First Class

Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$
```

2.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

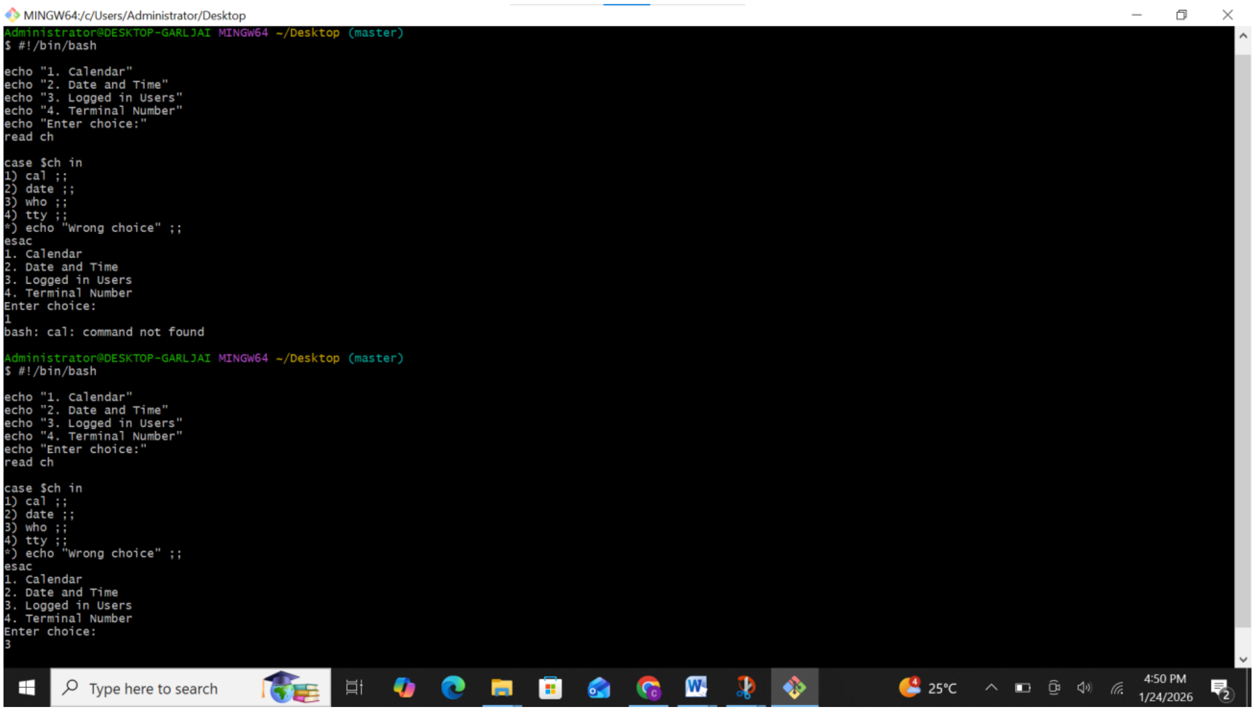
Code:

```
#!/bin/bash

echo "1. Calendar"

echo "2. Date and Time"

echo "3. Logged in Users"

echo "4. Terminal Number"

echo "Enter choice:"

read ch

case $ch in

1)   cal ;;
2)   date;
3)   who ;;
4)    tty ;;

*) echo "Wrong choice" ;;

esac
```

Output:



```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "1. Calendar"
echo "2. Date and Time"
echo "3. Logged in Users"
echo "4. Terminal Number"
echo "Enter choice:"
read ch

case $ch in
1) cal ;;
2) date ;;
3) who ;;
4) tty ;;
*) echo "Wrong choice" ;;
esac
1. Calendar
2. Date and Time
3. Logged in Users
4. Terminal Number
Enter choice:
2
Sat Jan 24 16:47:01 IST 2026

Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "1. Calendar"
echo "2. Date and Time"
echo "3. Logged in Users"
echo "4. Terminal Number"
echo "Enter choice:"
read ch

case $ch in
1) cal ;;
2) date ;;
3) who ;;
4) tty ;;
*) echo "Wrong choice" ;;
esac
1. Calendar
2. Date and Time
3. Logged in Users
4. Terminal Number
Enter choice:
4
/dev/pty0
```



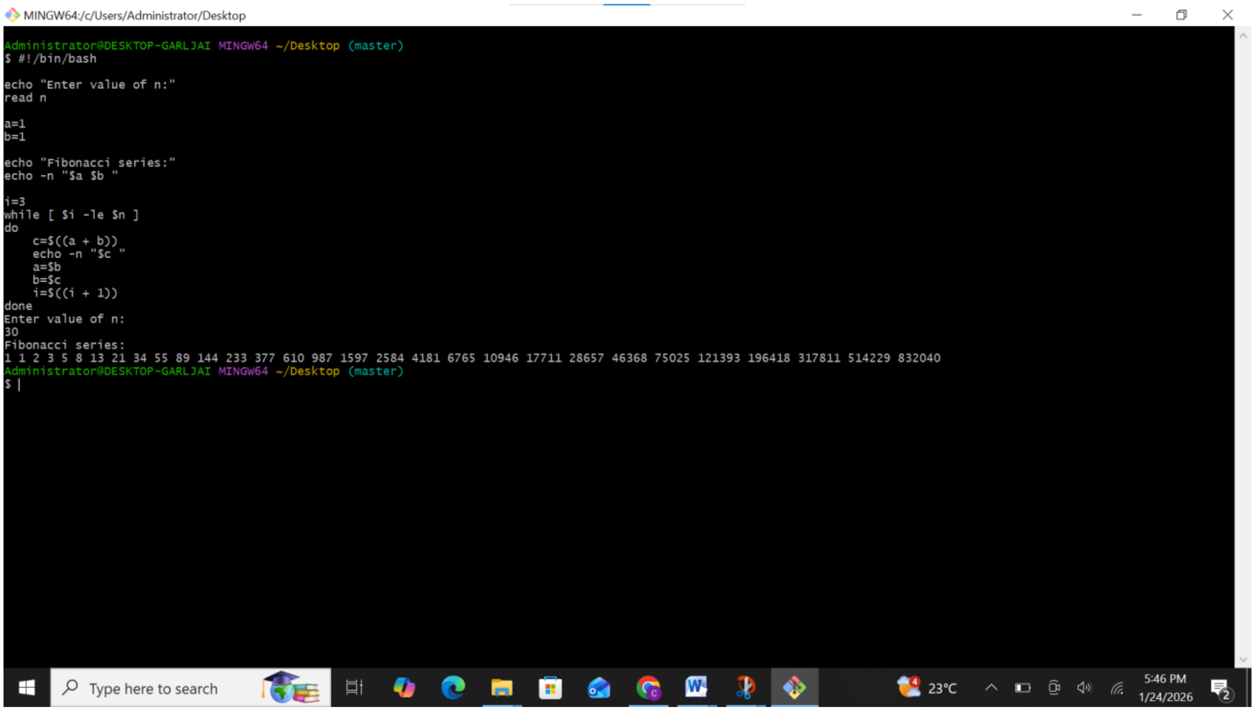
```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "1. Calendar"
echo "2. Date and Time"
echo "3. Logged in Users"
echo "4. Terminal Number"
echo "Enter choice:"
read ch

case $ch in
1) cal ;;
2) date ;;
3) who ;;
4) tty ;;
*) echo "Wrong choice" ;;
esac
1. Calendar
2. Date and Time
3. Logged in Users
4. Terminal Number
Enter choice:
1
bash: cal: command not found

Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "1. Calendar"
echo "2. Date and Time"
echo "3. Logged in Users"
echo "4. Terminal Number"
echo "Enter choice:"
read ch

case $ch in
1) cal ;;
2) date ;;
3) who ;;
4) tty ;;
*) echo "Wrong choice" ;;
esac
1. Calendar
2. Date and Time
3. Logged in Users
4. Terminal Number
Enter choice:
3
```

3.Write a shell script which will generate first n fibonnacci numbers like: 1, 1, 2, 3, 5, 13

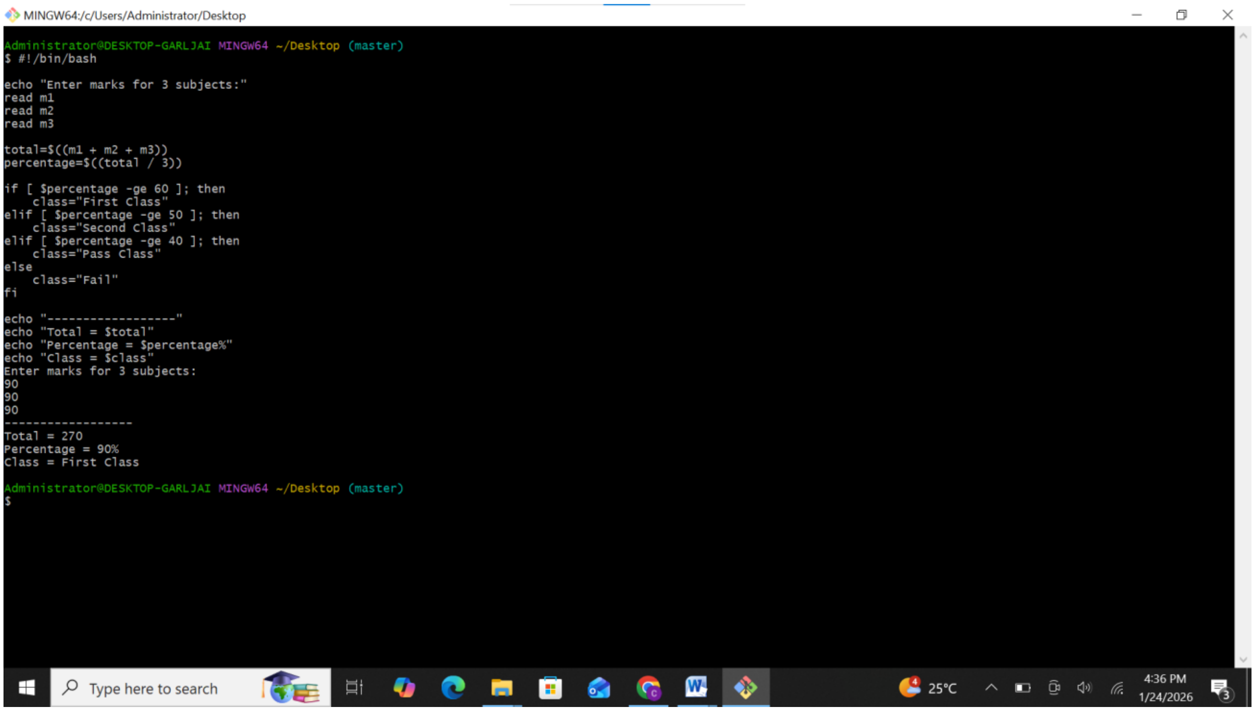
Code:

```
#!/bin/bash
echo "Enter value of n:"
read n
a=1
b=1
echo "Fibonacci series:"
echo -n "$a $b "
i=3
while [ $i -le $n ]
do
   c=$((a + b))
   echo -n "$c "
   a=$b
   b=$c
   i=$((i + 1))
done
```

Output:



```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "Enter value of n:"
read n

a=1
b=1

echo "Fibonacci series:"
echo -n "$a $b "

i=3
while [ $i -le $n ]
do
    c=$((a + b))
    echo -n "$c "
    a=$b
    b=$c
    i=$((i + 1))
done
Enter value of n:
30
Fibonacci series:
1 1 2 3 5 8 13 21 34 55 89 144 233 377 610 987 1597 2584 4181 6765 10946 17711 28657 46368 75025 121393 196418 317811 514229 832040
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ |
```



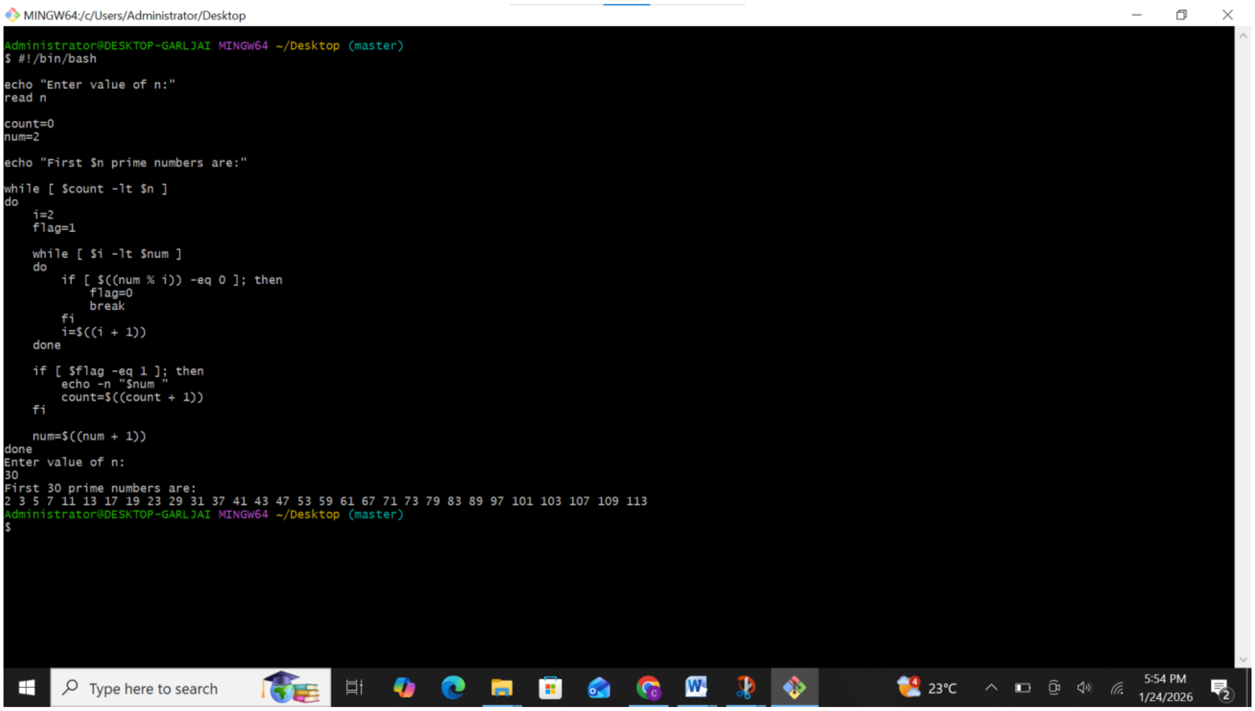
```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "Enter marks for 3 subjects:"
read m1
read m2
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 Class"
else
    class="Fail"
fi

echo "------------------"
echo "Total = $total"
echo "Percentage = $percentage%"
echo "Class = $class"
Enter marks for 3 subjects:
90
90
90
------------------
Total = 270
Percentage = 90%
Class = First Class

Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$
```

4.Write a shell script which will accept a number b and display first n prime numbers as output

Code:

```
#!/bin/bash
echo "Enter value of n:"
read n
count=0
num=2
echo "First $n prime numbers are:"
while [ $count -lt $n ]
do
    i=2
    flag=1
  while [ $i -lt $num ]
    do
        if [ $((num % i)) -eq 0 ]; then
            flag=0
            break
        fi
        i=$((i + 1))
    done
    if [ $flag -eq 1 ]; then
        echo -n "$num "
        count=$((count + 1))
    fi
 num=$((num + 1))
done
```

Output:



```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "Enter value of n:"
read n

count=0
num=2

echo "First $n prime numbers are:"

while [ $count -lt $n ]
do
    i=2
    flag=1

    while [ $i -lt $num ]
    do
        if [ $((num % i)) -eq 0 ]; then
            flag=0
            break
        fi
        i=$((i + 1))
    done

    if [ $flag -eq 1 ]; then
        echo -n "$num "
        count=$((count + 1))
    fi

    num=$((num + 1))
done
Enter value of n:
30
First 30 prime numbers are:
2 3 5 7 11 13 17 19 23 29 31 37 41 43 47 53 59 61 67 71 73 79 83 89 97 101 103 107 109 113
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$
```

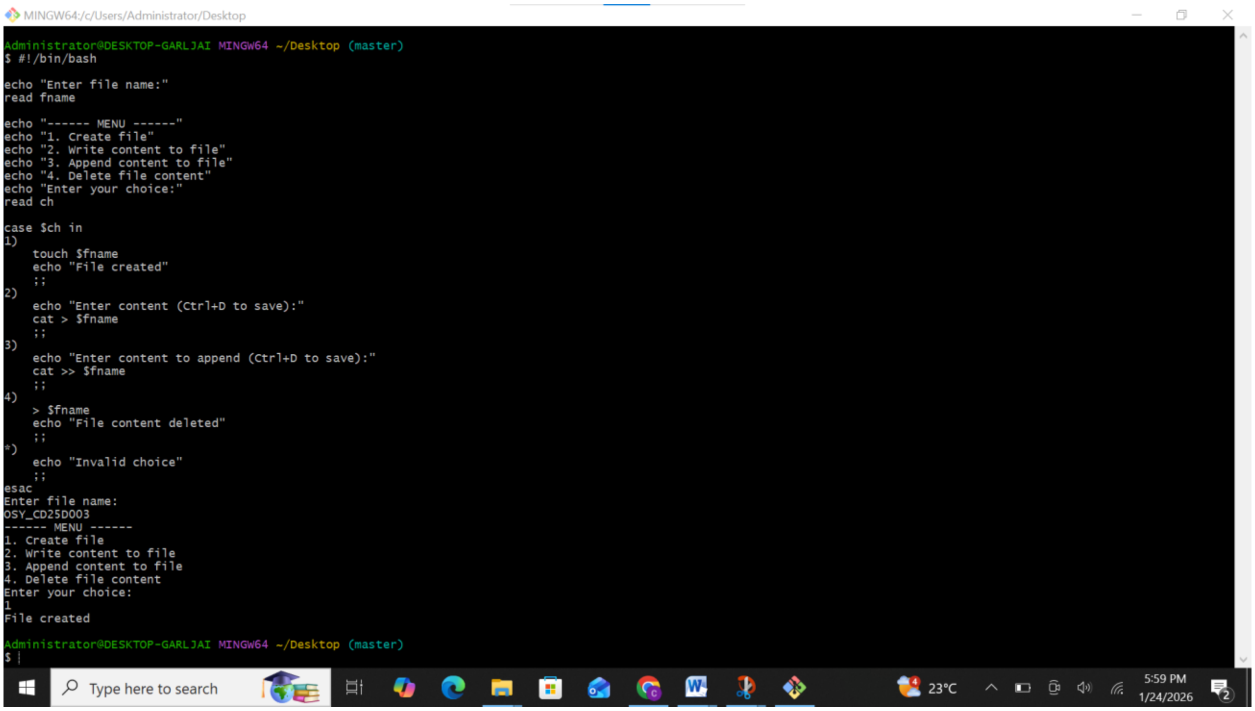
5.Write menu driven program for file handling activity ● Creation of file ● Write content in the file ● Upend file content ● Delete file content

Code:

```
#!/bin/bash

echo "Enter file name:"

read fname

echo "------ MENU ------"

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 ch

case $ch in
```

```
1) touch $fname

   echo "File created"

   ;;

2) echo "Enter content (Ctrl+D to save):"

   cat > $fname

   ;;

3)echo "Enter content to append (Ctrl+D to save):"

   cat >> $fname

   ;;

4) > $fname

   echo "File content deleted"

   ;;

*) echo "Invalid choice"

   ;;

Esac
```

Output:



```
Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$ #!/bin/bash

echo "Enter file name:"
read fname

echo "------ MENU ------"
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 ch

case $ch in
1)
    touch $fname
    echo "File created"
    ;;
2)
    echo "Enter content (Ctrl+D to save):"
    cat > $fname
    ;;
3)
    echo "Enter content to append (Ctrl+D to save):"
    cat >> $fname
    ;;
4)
    > $fname
    echo "File content deleted"
    ;;
*)
    echo "Invalid choice"
    ;;
esac
Enter file name:
OSY_CD25D003
------ MENU ------
1. Create file
2. Write content to file
3. Append content to file
4. Delete file content
Enter your choice:
1
File created

Administrator@DESKTOP-GARLJAI MINGW64 ~/Desktop (master)
$
```