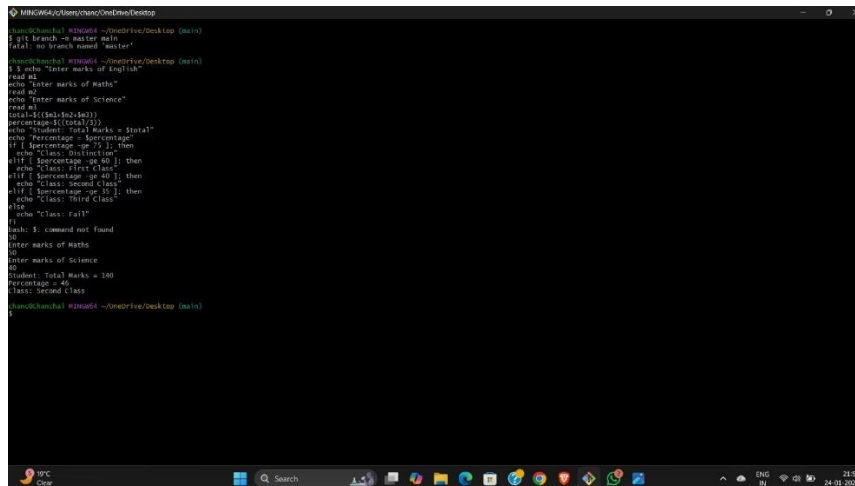


## PracticalNo : 2

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.

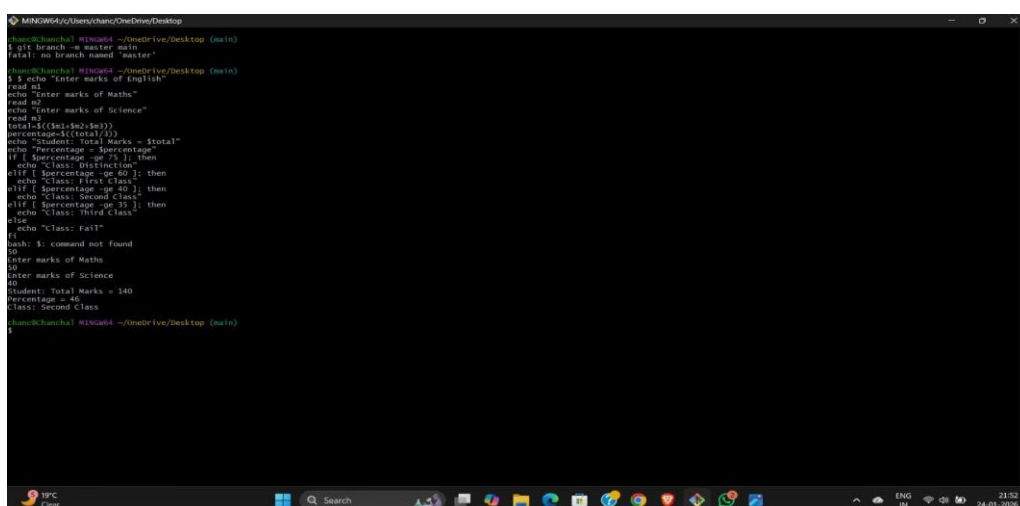


```
MINGW64~/Users/chanu/OneDrive/Desktop
chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$ git Branch -m master main
fatal: no branch named 'master'

chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$ $ echo "Enter marks of English"
read n1
echo "Enter marks of Maths"
read n2
echo "Enter marks of Science"
read n3
total=$((n1+n2+n3))
percentage=$((total/3))
echo "Student: Total Marks = $total"
echo "Percentage = $percentage"
if [ $percentage -ge 75 ]; then
echo "Class: Distinction"
elif [ $percentage -ge 60 ]; then
echo "Class: First Class"
elif [ $percentage -ge 40 ]; then
echo "Class: Second Class"
elif [ $percentage -ge 33 ]; then
echo "Class: Third Class"
else
echo "Class: Fail!"
fi
echo "Exit"
$ bash: $: command not found
$0
Enter marks of Maths
40
Enter marks of Science
50
Student: Total Marks = 140
Percentage = 40
Class: Second Class
chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$
```

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

- I. Display calendar of current month
- II. Display today's date and time
- III. Display usernames those are currently logged in the system
- IV. Display your terminal number



```
MINGW64~/Users/chanu/OneDrive/Desktop
chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$ git Branch -m master main
fatal: no branch named 'master'

chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$ $ echo "Enter marks of English"
read n1
echo "Enter marks of Maths"
read n2
echo "Enter marks of Science"
read n3
total=$((n1+n2+n3))
percentage=$((total/3))
echo "Student: Total Marks = $total"
echo "Percentage = $percentage"
if [ $percentage -ge 75 ]; then
echo "Class: Distinction"
elif [ $percentage -ge 60 ]; then
echo "Class: First Class"
elif [ $percentage -ge 40 ]; then
echo "Class: Second Class"
elif [ $percentage -ge 33 ]; then
echo "Class: Third Class"
else
echo "Class: Fail!"
fi
echo "Exit"
$ bash: $: command not found
$0
Enter marks of Maths
40
Enter marks of Science
50
Student: Total Marks = 140
Percentage = 40
Class: Second Class
chanchanchal@MINGW64 ~/OneDrive/Desktop (main)
$
```

```
MINGW64/c/Users/chanc/OneDrive/Desktop
chanc@chancha1 MINGW64 ~/OneDrive/Desktop (main)
$ $ #!/bin/bash

echo "1. Calendar of current month"
echo "2. Today's date and time"
echo "3. Logged in users"
echo "4. Terminal number"
echo "Enter your choice"
read ch

if [ $ch -eq 1 ]; then
    date +%B %Y
elif [ $ch -eq 2 ]; then
    date
elif [ $ch -eq 3 ]; then
    who
elif [ $ch -eq 4 ]; then
    tty
else
    echo "Invalid choice"
fi
bash: $: command not found
1. Calendar of current month
2. Today's date and time
3. Logged in users
4. Terminal number
Enter your choice
2
Sat Jan 24 22:02:38 IST 2026
```

```
chanc@chancha1 MINGW64 ~/OneDrive/Desktop (main)
$ $ #!/bin/bash

echo "1. Calendar of current month"
echo "2. Today's date and time"
echo "3. Logged in users"
echo "4. Terminal number"
echo "Enter your choice"
read ch

if [ $ch -eq 1 ]; then
    date +%B %Y
elif [ $ch -eq 2 ]; then
    date
elif [ $ch -eq 3 ]; then
    who
elif [ $ch -eq 4 ]; then
    tty
else
    echo "Invalid choice"
fi
bash: $: command not found
1. Calendar of current month
2. Today's date and time
3. Logged in users
4. Terminal number
Enter your choice
4
/dev/pty0
chanc@chancha1 MINGW64 ~/OneDrive/Desktop (main)
$
```

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

```
MINGW64/c/Users/chanc/OneDrive/Desktop
chanc@chancha1 MINGW64 ~/OneDrive/Desktop (main)
$ #!/bin/bash

echo "Enter how many Fibonacci numbers you want"
read n

a=1
b=1

echo "Fibonacci Series:"
if [ $n -ge 2 ]; then
    printf "%d\n" $a
fi

for ((i=2; i<=n; i++))
do
    c=$((a + b))
    printf "%d\n" $c
    a=b
    b=c
done

echo
echo "Enter how many Fibonacci numbers you want"
n=13
echo "Fibonacci Series:"
1 1 2 3 5 8 13
```

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

```
MINGW64/c/Users/chanc/OneDrive/Desktop
chanc@chancha1 MINGW64 ~/OneDrive/Desktop (main)
$ #!/bin/bash

echo "Enter the value of n"
read n
count=0
num=2
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
echo "Enter the value of n"
9
First 9 prime numbers are:
2 3 5 7 11 13 17 19 23
```

## 5. Write menu driven program for file handling activity

- I. Creation of file
- II. Write content in the file
- III. Upend file content
- IV. Delete file content

```
MINGW64/c/Users/chanc/OneDrive/Desktop
chanc@chanchal MINGW64 ~/OneDrive/Desktop (main)
$ #!/bin/bash
echo "1) Create File"
echo "2) write Content"
echo "3) Append Content"
echo "4) Delete File Content"
echo "Enter choice:"
read ch
echo "Enter file name:"
read fname
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
1) Create File
2) write Content
3) Append Content
4) Delete File Content
Enter choice:
1
Enter file name:
chanchal(CD24013)
File created
```