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.

```
VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ touch practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ chmod +x practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano prctical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical1.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical1.sh
Enter marks of subject 1:
80
Enter marks of subject 2:
87
Enter marks of subject 3:
70
Total Marks: 237
Percentage: 79%
Class: Distinction
```

```
MINGW64:/c/Users/VICTUS/shell_practical
  GNU nano 8.7                                practical1.sh
#!/bin/bash

echo "Enter marks of subject 1:"
read m1
echo "Enter marks of subject 2:"
read m2
echo "Enter marks of subject 3:"
read m3

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

echo "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: Pass"
else
  echo "Class: Fail"
fi
```

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

```
VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ touch practical2.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical2.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ chmod +x practical2.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical2.sh
1. Display calendar of current month
2. Display today's date and time
3. Display usernames currently logged in
4. Display terminal number
Enter your choice:
2
Mon Jan 26 20:33:24 IST 2026

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$
```

```
MINGW64:/c/Users/VICTUS/shell_practical
  GNU nano 8.7                    practical2.sh                    Modified
#!/bin/bash

echo "1. Display calendar of current month"
echo "2. Display today's date and time"
echo "3. Display usernames currently logged in"
echo "4. Display terminal number"
echo "Enter your choice:"
read ch

case $ch in
1) cal ;;
2) date ;;
3) who ;;
4) tty ;;
*) echo "Invalid choice" ;;
esac

Write to File: practical2.sh
^G Help          M-D DOS Format      M-A Append          M-B Backup File     AT Browse
^C Cancel        M-M Mac Format      M-P Prepend         ^Q Discard buffer
```

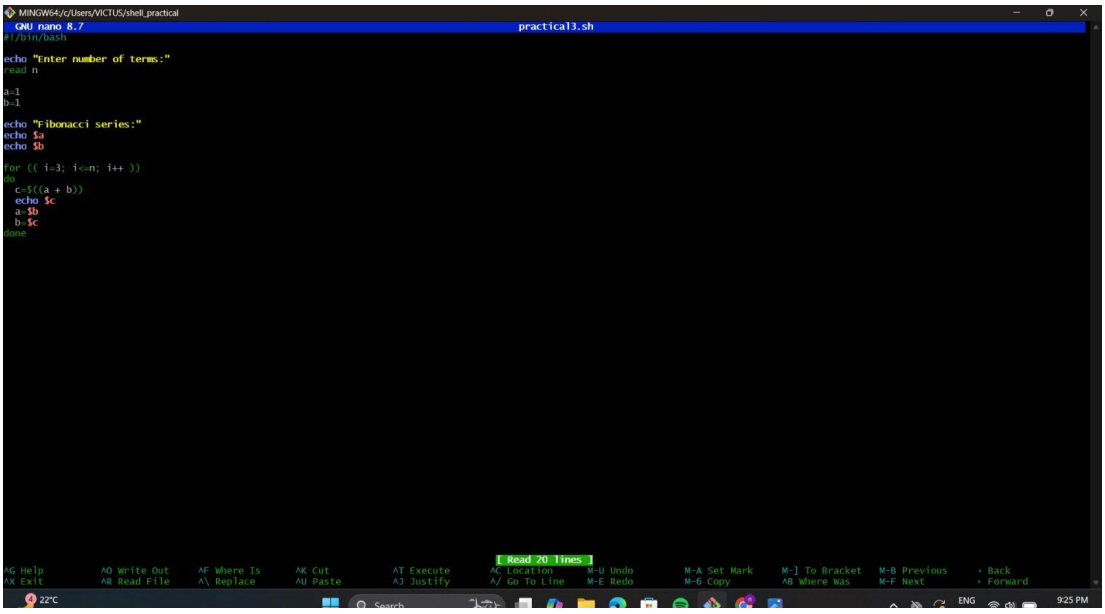
3. Write a shell script which will generate first n Fibonacci numbers like: 1, 1,2, 3, 5, 13

```
VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ touch practical3.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical3.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ chmod +x practical3.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical3.sh
Enter number of terms:
6
Fibonacci series:
1
1
2
3
5
8
```



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

```
VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ touch practical4.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical4.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ chmod +x practical4.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical4.sh
Enter how many prime numbers:
3
2
3
5
```

MINGW64:/c/Users/VICTUS/shell_practical

GNU nano 8.7 practical4.sh

```
#!/bin/bash

echo "Enter how many prime numbers:"
read n

count=0
num=2

while [ $count -lt $n ]
do
  flag=1
  for (( i=2; i<=num/2; i++ ))
  do
    if [ $((num % i)) -eq 0 ]
    then
      flag=0
      break
    fi
  done

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

  num=$((num + 1))
done
```

5. Write menu driven program for file handling activity

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

```
5

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ touch practical5,sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ nano practical5.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ chmod +x practical5.sh

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ,/practical5.sh
bash: ,/practical5.sh: No such file or directory

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$ ./practical5.sh
1. Create file
2. Write to file
3. Append to file
4. Delete file content
Enter your choice:
5
Enter filename:
test.txt
Invalid choice

VICTUS@DESKTOP-OMT7HC7 MINGW64 ~/shell_practical
$
```

```
MINGW64:/c/Users/VICTUS/shell_practical
  GNU nano 8.7                                practical5.sh
#!/bin/bash

echo "1. Create file"
echo "2. Write to file"
echo "3. Append to file"
echo "4. Delete file content"
echo "Enter your choice:"
read ch

echo "Enter filename:"
read fname

case $ch in
1)
  touch $fname
  echo "File created"
  ;;
2)
  echo "Enter content:"
  read content
  echo "$content" > $fname
  ;;
3)
  echo "Enter content:"
  read content
  echo "$content" >> $fname
  ;;
4)
  > $fname
  echo "File content deleted"
  ;;
*)
  echo "Invalid choice"
  ;;
esac

                              [ Read 35 lines ]
^G Help        ^O Write Out   ^F Where Is    ^K Cut         ^T Execute     ^C Location    M-U Undo       M-A Set Mark   M-] To Bracket  M-B Previous   ^← Back
^X Exit        ^R Read File   ^\ Replace     ^U Paste       ^J Justify     ^/ Go To Line  M-E Redo       M-6 Copy       ^B Where Was    M-F Next       ^→ Forward
```