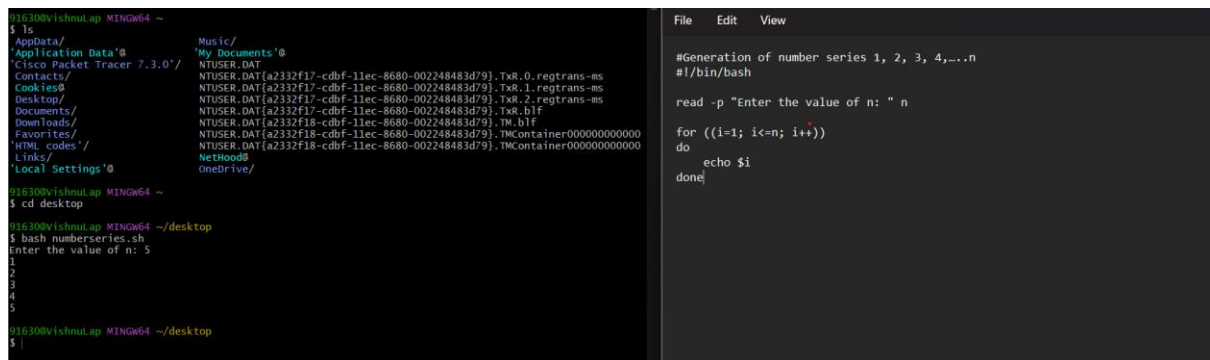


## Exp-1

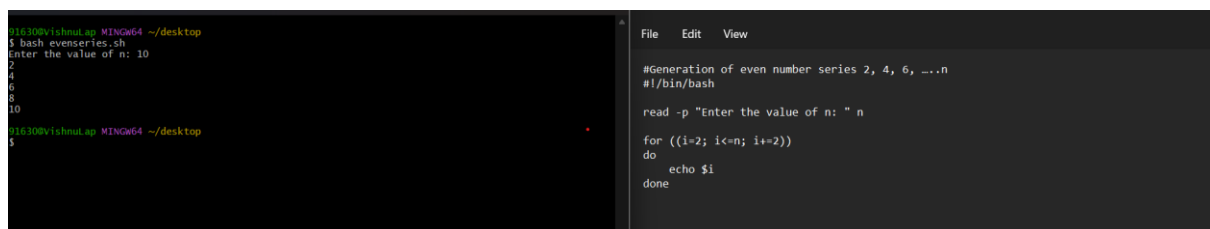


The screenshot shows a Windows terminal window with a file explorer in the background. The terminal displays the execution of a script named 'numberseries.sh'. The user enters '5' for the value of 'n', and the script outputs the series 1, 2, 3, 4, 5. The Notepad window shows the script code, which uses a 'for' loop to generate the series.

```
016300VishnuLap MINGW64 ~  
$ ls  
AppData/ Music/  
Application Data/ My Documents/  
Cisco Packet Tracer 7.3.0/ NTUSER.DAT  
Contacts/ NTUSER.DAT{a2332f17-cdbf-11ec-8680-002248483d79}.TxR.0.regtrans-ms  
Cookies/ NTUSER.DAT{a2332f17-cdbf-11ec-8680-002248483d79}.TxR.1.regtrans-ms  
Desktop/ NTUSER.DAT{a2332f17-cdbf-11ec-8680-002248483d79}.TxR.2.regtrans-ms  
Documents/ NTUSER.DAT{a2332f17-cdbf-11ec-8680-002248483d79}.TxR.b1f  
Downloads/ NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TM.b1f  
Favorites/ NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TMCContainer000000000000  
HTML codes/ NTUSER.DAT{a2332f18-cdbf-11ec-8680-002248483d79}.TMCContainer000000000000  
Links/ NetHood  
Local Settings/ OneDrive/  
016300VishnuLap MINGW64 ~  
$ cd desktop  
016300VishnuLap MINGW64 ~/desktop  
$ bash numberseries.sh  
Enter the value of n: 5  
1  
2  
3  
4  
5  
016300VishnuLap MINGW64 ~/desktop  
$
```

```
File Edit View  
#Generation of number series 1, 2, 3, 4,...n  
#!/bin/bash  
  
read -p "Enter the value of n: " n  
  
for ((i=1; i<=n; i++))  
do  
    echo $i  
done
```

## Exp-2

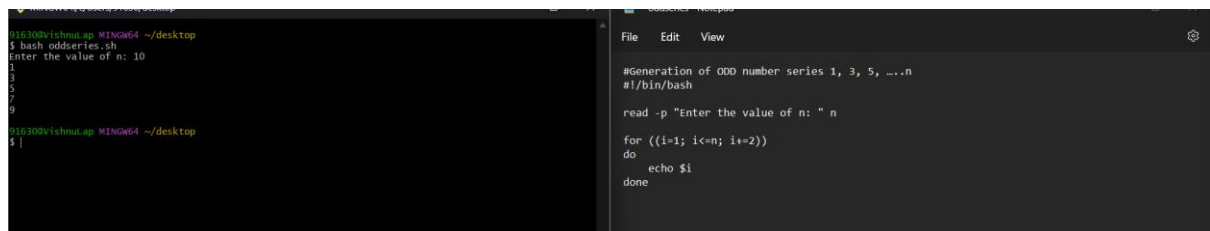


The screenshot shows a Windows terminal window with a file explorer in the background. The terminal displays the execution of a script named 'evenseries.sh'. The user enters '10' for the value of 'n', and the script outputs the even series 2, 4, 6, 8, 10. The Notepad window shows the script code, which uses a 'for' loop to generate the series.

```
016300VishnuLap MINGW64 ~/desktop  
$ bash evenseries.sh  
Enter the value of n: 10  
2  
4  
6  
8  
10  
016300VishnuLap MINGW64 ~/desktop  
$
```

```
File Edit View  
#Generation of even number series 2, 4, 6, ...n  
#!/bin/bash  
  
read -p "Enter the value of n: " n  
  
for ((i=2; i<=n; i+=2))  
do  
    echo $i  
done
```

## Exp-3

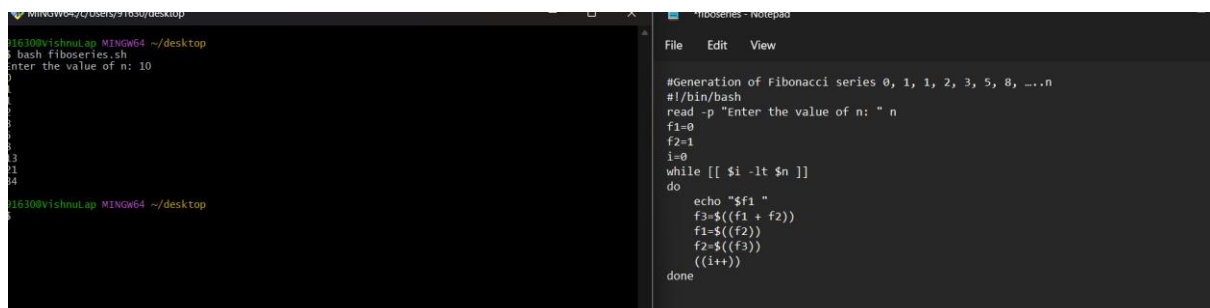


The screenshot shows a Windows terminal window with a file explorer in the background. The terminal displays the execution of a script named 'oddsseries.sh'. The user enters '10' for the value of 'n', and the script outputs the odd series 1, 3, 5, 7, 9. The Notepad window shows the script code, which uses a 'for' loop to generate the series.

```
016300VishnuLap MINGW64 ~/desktop  
$ bash oddsseries.sh  
Enter the value of n: 10  
1  
3  
5  
7  
9  
016300VishnuLap MINGW64 ~/desktop  
$
```

```
File Edit View  
#Generation of ODD number series 1, 3, 5, ...n  
#!/bin/bash  
  
read -p "Enter the value of n: " n  
  
for ((i=1; i<=n; i+=2))  
do  
    echo $i  
done
```

## Exp-4

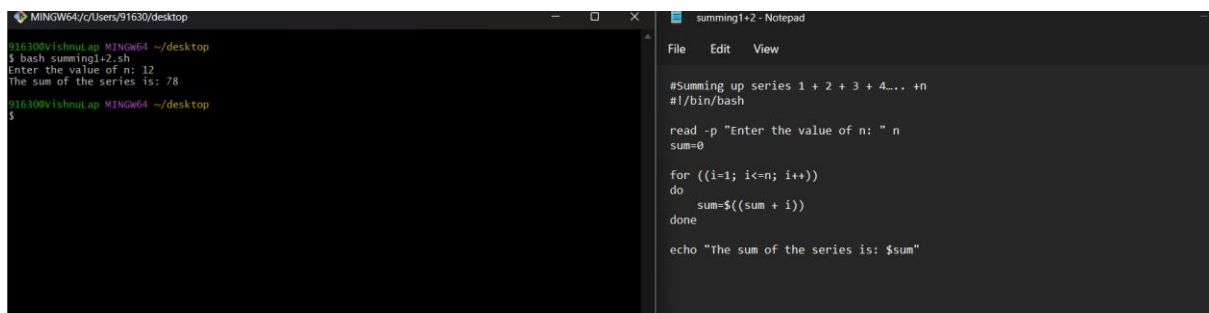


The screenshot shows a Windows terminal window with a file explorer in the background. The terminal displays the execution of a script named 'fiboseries.sh'. The user enters '10' for the value of 'n', and the script outputs the Fibonacci series 0, 1, 1, 2, 3, 5, 8. The Notepad window shows the script code, which uses a 'while' loop to generate the series.

```
016300VishnuLap MINGW64 ~/desktop  
$ bash fiboseries.sh  
Enter the value of n: 10  
0  
1  
1  
2  
3  
5  
8  
13  
21  
34  
016300VishnuLap MINGW64 ~/desktop  
$
```

```
File Edit View  
#Generation of Fibonacci series 0, 1, 1, 2, 3, 5, 8, ...n  
#!/bin/bash  
read -p "Enter the value of n: " n  
f1=0  
f2=1  
i=0  
while [[ $i -lt $n ]]  
do  
    echo "$f1 "  
    f3=$((f1 + f2))  
    f1=$((f2))  
    f2=$((f3))  
    ((i++))  
done
```

## Exp-5



The screenshot shows a terminal window on the left and a Notepad window on the right. The terminal window title is "MINGW64/c/Users/91630/desktop". It shows the execution of a script named "summing1+2.sh". The user enters "12" for the value of n, and the script outputs "The sum of the series is: 78". The Notepad window title is "summing1+2 - Notepad". It contains the following bash script:

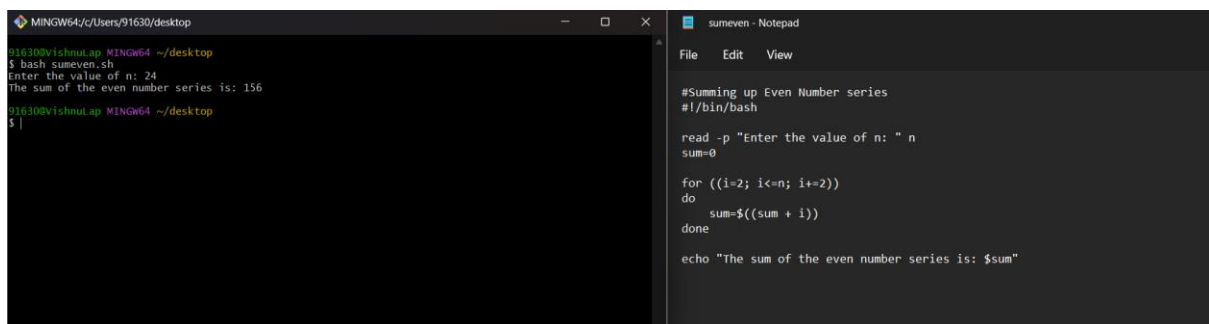
```
#Summing up series 1 + 2 + 3 + 4... +n
#!/bin/bash

read -p "Enter the value of n: " n
sum=0

for ((i=1; i<=n; i++))
do
    sum=$((sum + i))
done

echo "The sum of the series is: $sum"
```

## Exp-6



The screenshot shows a terminal window on the left and a Notepad window on the right. The terminal window title is "MINGW64/c/Users/91630/desktop". It shows the execution of a script named "sumeven.sh". The user enters "24" for the value of n, and the script outputs "The sum of the even number series is: 156". The Notepad window title is "sumeven - Notepad". It contains the following bash script:

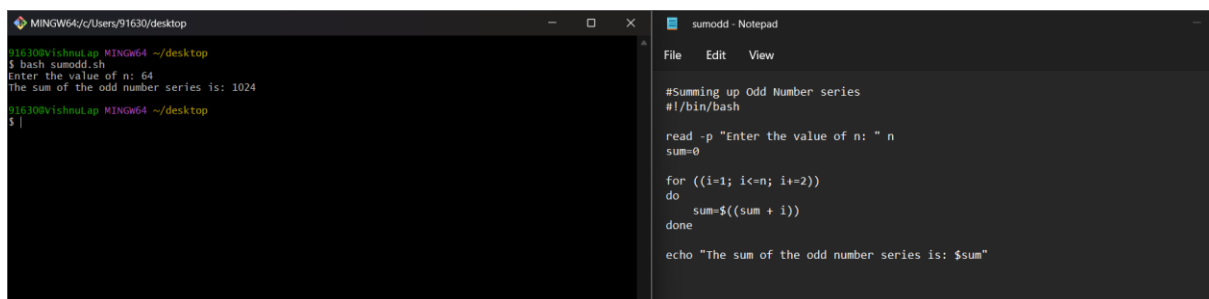
```
#Summing up Even Number series
#!/bin/bash

read -p "Enter the value of n: " n
sum=0

for ((i=2; i<=n; i+=2))
do
    sum=$((sum + i))
done

echo "The sum of the even number series is: $sum"
```

## Exp-7



The screenshot shows a terminal window on the left and a Notepad window on the right. The terminal window title is "MINGW64/c/Users/91630/desktop". It shows the execution of a script named "sumodd.sh". The user enters "64" for the value of n, and the script outputs "The sum of the odd number series is: 1024". The Notepad window title is "sumodd - Notepad". It contains the following bash script:

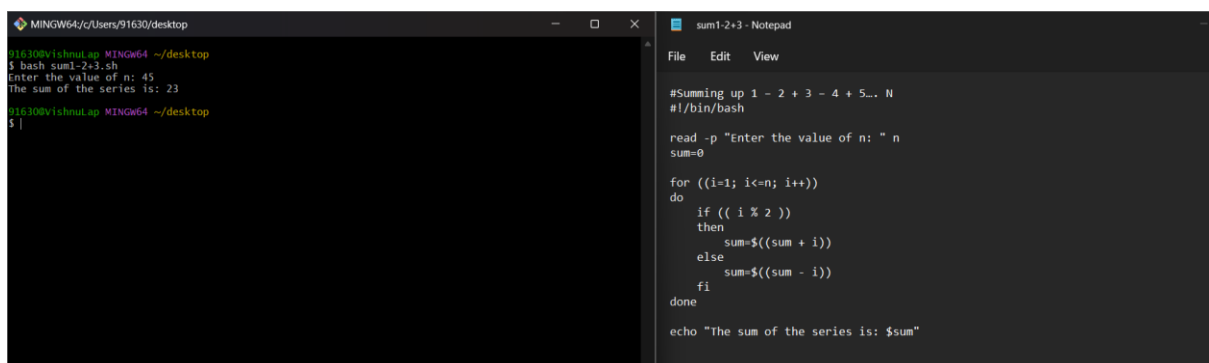
```
#Summing up Odd Number series
#!/bin/bash

read -p "Enter the value of n: " n
sum=0

for ((i=1; i<=n; i+=2))
do
    sum=$((sum + i))
done

echo "The sum of the odd number series is: $sum"
```

## Exp-8



The screenshot shows a terminal window on the left and a Notepad window on the right. The terminal window title is "MINGW64/c/Users/91630/desktop". It shows the execution of a script named "sum1-2+3.sh". The user enters "45" for the value of n, and the script outputs "The sum of the series is: 23". The Notepad window title is "sum1-2+3 - Notepad". It contains the following bash script:

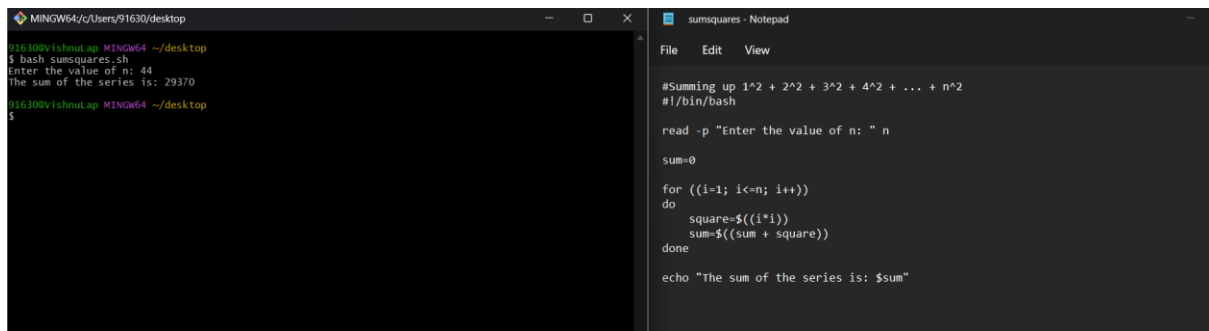
```
#Summing up 1 - 2 + 3 - 4 + 5... N
#!/bin/bash

read -p "Enter the value of n: " n
sum=0

for ((i=1; i<=n; i++))
do
    if (( i % 2 ))
    then
        sum=$((sum + i))
    else
        sum=$((sum - i))
    fi
done

echo "The sum of the series is: $sum"
```

## Exp-9



The screenshot shows a terminal window on the left and a Notepad window titled 'sumsquares - Notepad' on the right. The terminal shows the execution of a script named 'sumsquares.sh' where the user enters '44' and the output is 'The sum of the series is: 29370'. The Notepad window contains the source code of the script, which calculates the sum of squares from 1 to n using a for loop.

```
MINGW64/c/Users/91630/desktop
91630@VishnuLap MINGW64 ~/desktop
$ bash sumsquares.sh
Enter the value of n: 44
The sum of the series is: 29370
91630@VishnuLap MINGW64 ~/desktop
$
```

```
sumsquares - Notepad
File Edit View

#Summing up 1^2 + 2^2 + 3^2 + 4^2 + ... + n^2
#!/bin/bash

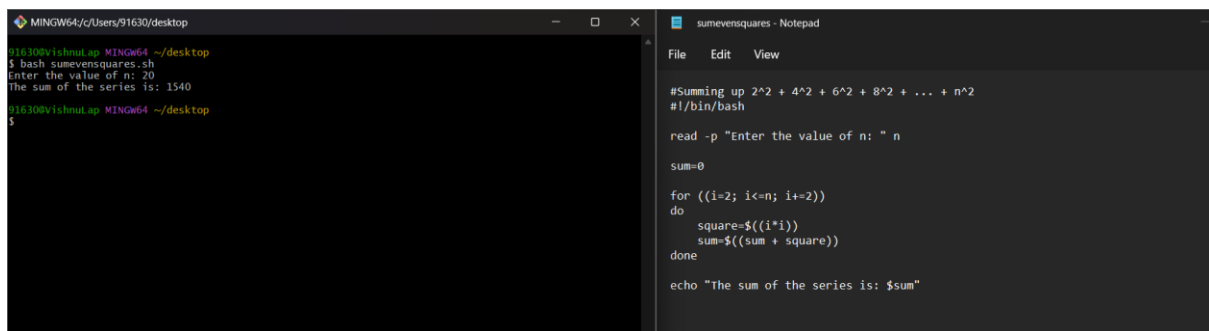
read -p "Enter the value of n: " n

sum=0

for ((i=1; i<=n; i++))
do
    square=$((i*i))
    sum=$((sum + square))
done

echo "The sum of the series is: $sum"
```

## Exp-10



The screenshot shows a terminal window on the left and a Notepad window titled 'sumeversquares - Notepad' on the right. The terminal shows the execution of a script named 'sumeversquares.sh' where the user enters '20' and the output is 'The sum of the series is: 1540'. The Notepad window contains the source code of the script, which calculates the sum of even squares from 2 to n using a for loop.

```
MINGW64/c/Users/91630/desktop
91630@VishnuLap MINGW64 ~/desktop
$ bash sumevensquares.sh
Enter the value of n: 20
The sum of the series is: 1540
91630@VishnuLap MINGW64 ~/desktop
$
```

```
sumeversquares - Notepad
File Edit View

#Summing up 2^2 + 4^2 + 6^2 + 8^2 + ... + n^2
#!/bin/bash

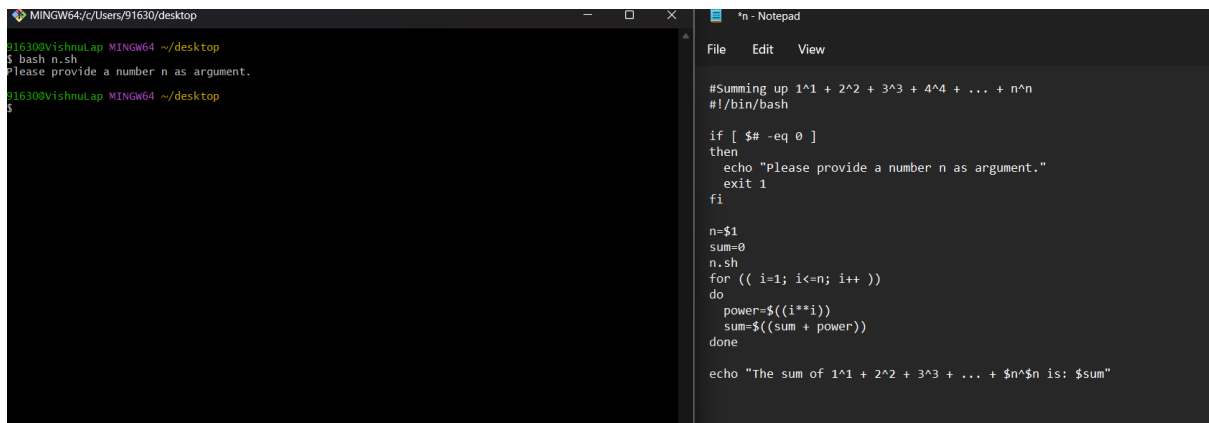
read -p "Enter the value of n: " n

sum=0

for ((i=2; i<=n; i+=2))
do
    square=$((i*i))
    sum=$((sum + square))
done

echo "The sum of the series is: $sum"
```

## Exp-11



The screenshot shows a terminal window on the left and a Notepad window titled '\*n - Notepad' on the right. The terminal shows the execution of a script named 'n.sh' where the user provides the argument '4' and the output is 'The sum of 1^1 + 2^2 + 3^3 + ... + 4^4 is: 100'. The Notepad window contains the source code of the script, which calculates the sum of powers from 1 to n using a for loop.

```
MINGW64/c/Users/91630/desktop
91630@VishnuLap MINGW64 ~/desktop
$ bash n.sh
Please provide a number n as argument.
91630@VishnuLap MINGW64 ~/desktop
$
```

```
*n - Notepad
File Edit View

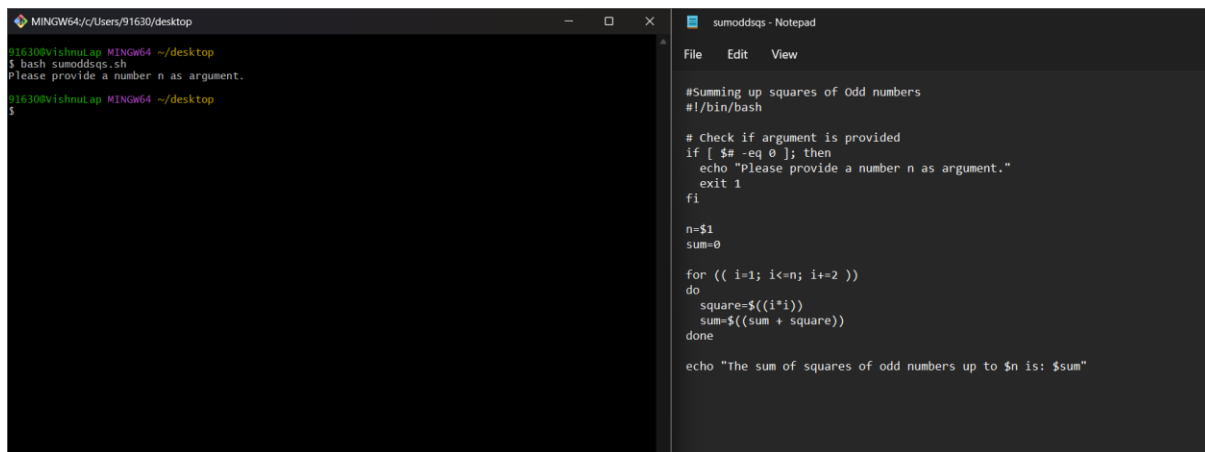
#Summing up 1^1 + 2^2 + 3^3 + 4^4 + ... + n^n
#!/bin/bash

if [ $# -eq 0 ]
then
    echo "Please provide a number n as argument."
    exit 1
fi

n=$1
sum=0
n.sh
for (( i=1; i<=n; i++ ))
do
    power=$((i**i))
    sum=$((sum + power))
done

echo "The sum of 1^1 + 2^2 + 3^3 + ... + $n^n is: $sum"
```

## Exp-12



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'sumoddsqs - Notepad'. The terminal shows the execution of a script named 'sumoddsqs.sh' which prompts the user for a number 'n'. The Notepad window displays the source code of the script.

```
91630@vishnuLap MINGW64 ~/desktop
$ bash sumoddsqs.sh
Please provide a number n as argument.
91630@vishnuLap MINGW64 ~/desktop
$
```

```
#Summing up squares of Odd numbers
#!/bin/bash

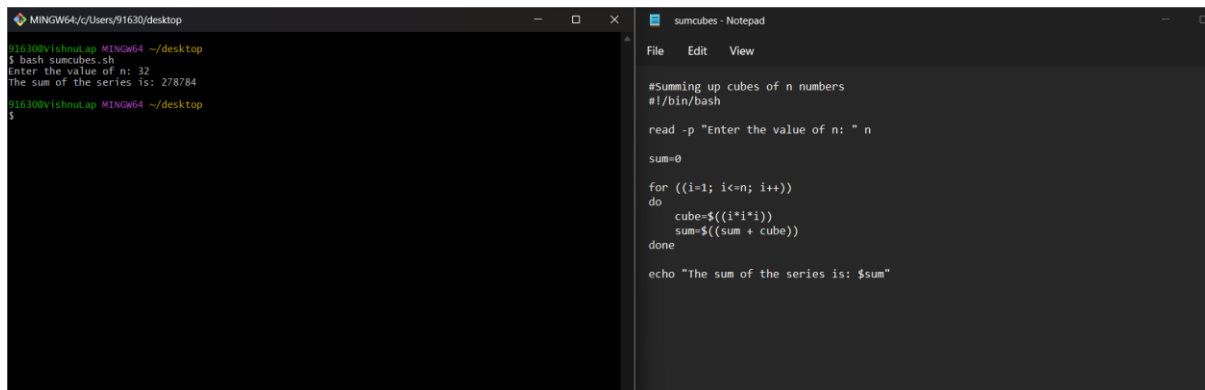
# Check if argument is provided
if [ $# -eq 0 ]; then
    echo "Please provide a number n as argument."
    exit 1
fi

n=$1
sum=0

for (( i=1; i<=n; i+=2 ))
do
    square=$((i*i))
    sum=$((sum + square))
done

echo "The sum of squares of odd numbers up to $n is: $sum"
```

## Exp-13



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'sumcubes - Notepad'. The terminal shows the execution of a script named 'sumcubes.sh' which prompts the user for a value 'n'. The Notepad window displays the source code of the script.

```
91630@vishnuLap MINGW64 ~/desktop
$ bash sumcubes.sh
Enter the value of n: 32
The sum of the series is: 278784
91630@vishnuLap MINGW64 ~/desktop
$
```

```
#Summing up cubes of n numbers
#!/bin/bash

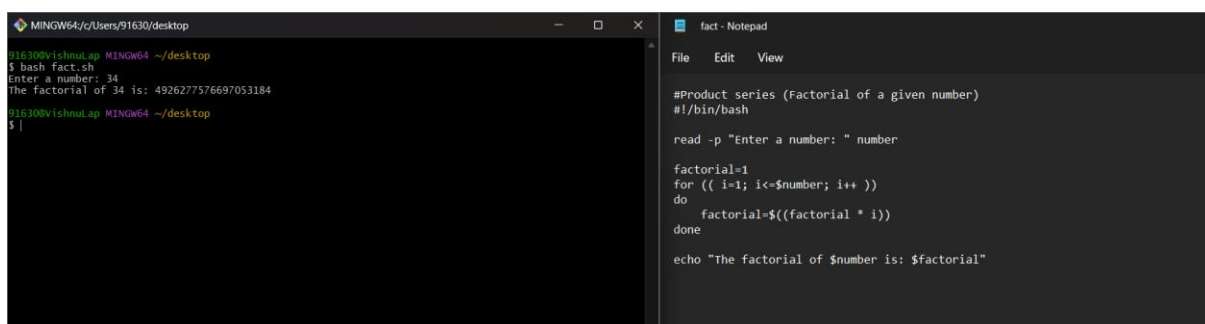
read -p "Enter the value of n: " n

sum=0

for ((i=1; i<=n; i++))
do
    cube=$((i*i*i))
    sum=$((sum + cube))
done

echo "The sum of the series is: $sum"
```

## Exp-14



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'fact - Notepad'. The terminal shows the execution of a script named 'fact.sh' which prompts the user for a number. The Notepad window displays the source code of the script.

```
91630@vishnuLap MINGW64 ~/desktop
$ bash fact.sh
Enter a number: 34
The factorial of 34 is: 4926277576697053184
91630@vishnuLap MINGW64 ~/desktop
$
```

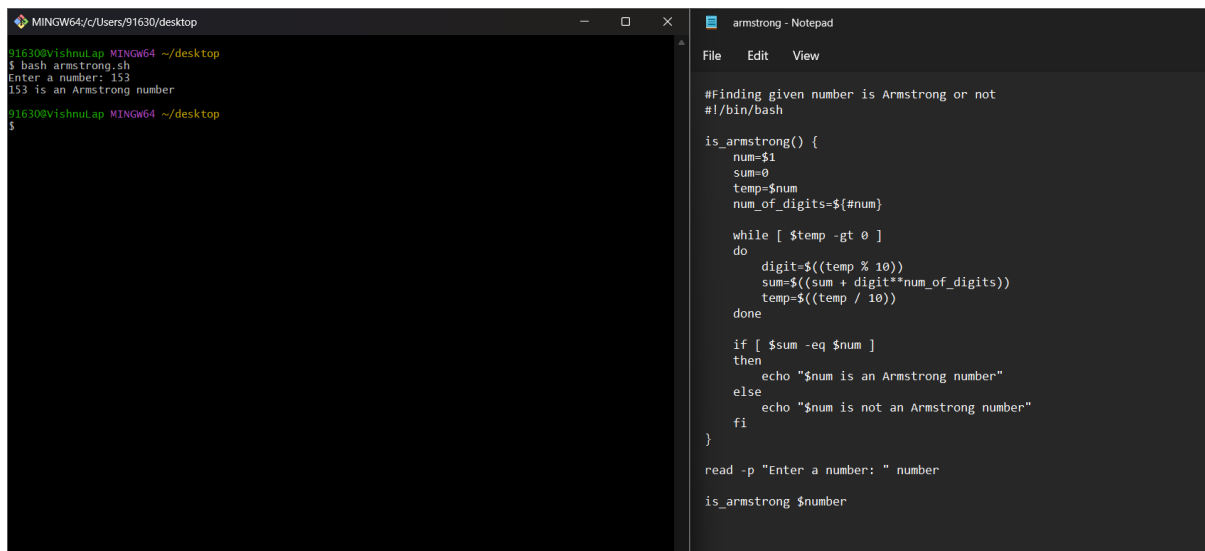
```
#Product series (Factorial of a given number)
#!/bin/bash

read -p "Enter a number: " number

factorial=1
for (( i=1; i<=number; i++ ))
do
    factorial=$((factorial * i))
done

echo "The factorial of $number is: $factorial"
```

## Exp-15



The screenshot shows a terminal window on the left and a Notepad file named 'armstrong - Notepad' on the right. The terminal shows the execution of a script 'armstrong.sh' which checks if the number 153 is an Armstrong number. The output is '153 is an Armstrong number'. The Notepad file contains the following code:

```
#Finding given number is Armstrong or not
#!/bin/bash

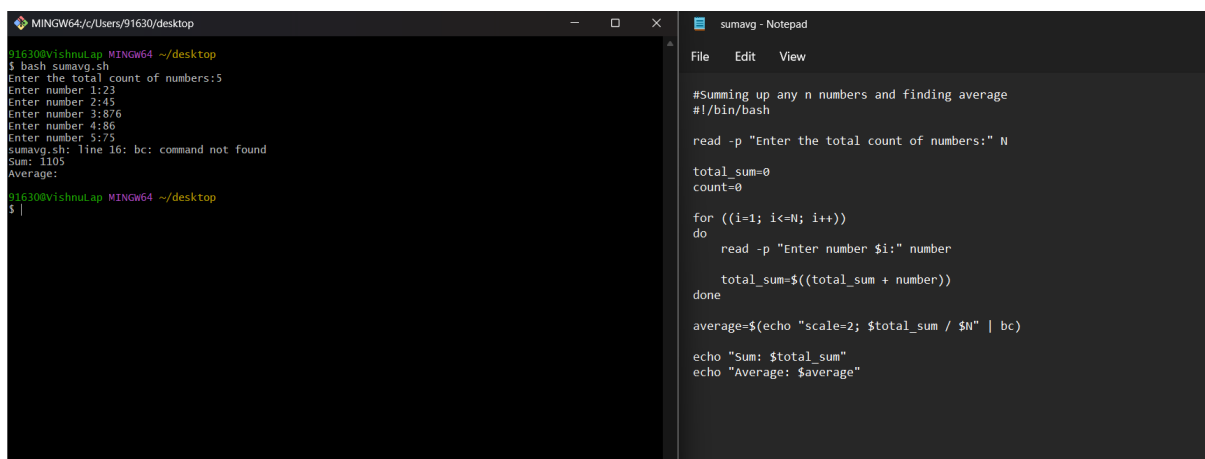
is_armstrong() {
    num=$1
    sum=0
    temp=$num
    num_of_digits=${#num}

    while [ $temp -gt 0 ]
    do
        digit=$((temp % 10))
        sum=$((sum + digit**num_of_digits))
        temp=$((temp / 10))
    done

    if [ $sum -eq $num ]
    then
        echo "$num is an Armstrong number"
    else
        echo "$num is not an Armstrong number"
    fi
}

read -p "Enter a number: " number
is_armstrong $number
```

## Exp-16



The screenshot shows a terminal window on the left and a Notepad file named 'sumavg - Notepad' on the right. The terminal shows the execution of a script 'sumavg.sh' which prompts the user to enter the total count of numbers (5) and then five numbers (1, 2, 3, 4, 5). The output is 'Sum: 1105' and 'Average: 221'. The Notepad file contains the following code:

```
#Summing up any n numbers and finding average
#!/bin/bash

read -p "Enter the total count of numbers:" N

total_sum=0
count=0

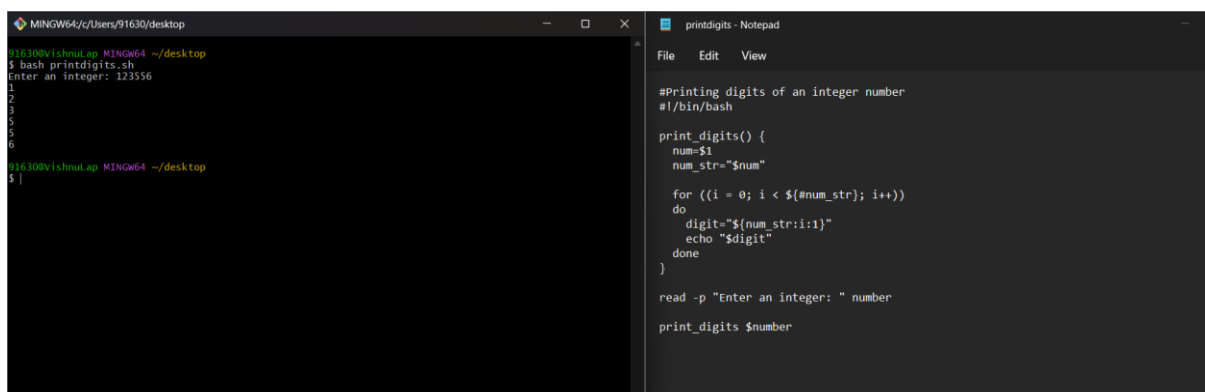
for ((i=1; i<=N; i++))
do
    read -p "Enter number $i:" number

    total_sum=$((total_sum + number))
done

average=$((echo "scale=2; $total_sum / $N" | bc))

echo "Sum: $total_sum"
echo "Average: $average"
```

## Exp-17



The screenshot shows a terminal window on the left and a Notepad file named 'printdigits - Notepad' on the right. The terminal shows the execution of a script 'printdigits.sh' which prompts the user to enter an integer (123556) and then prints each digit on a new line. The output is '1', '2', '3', '5', '5', '6'. The Notepad file contains the following code:

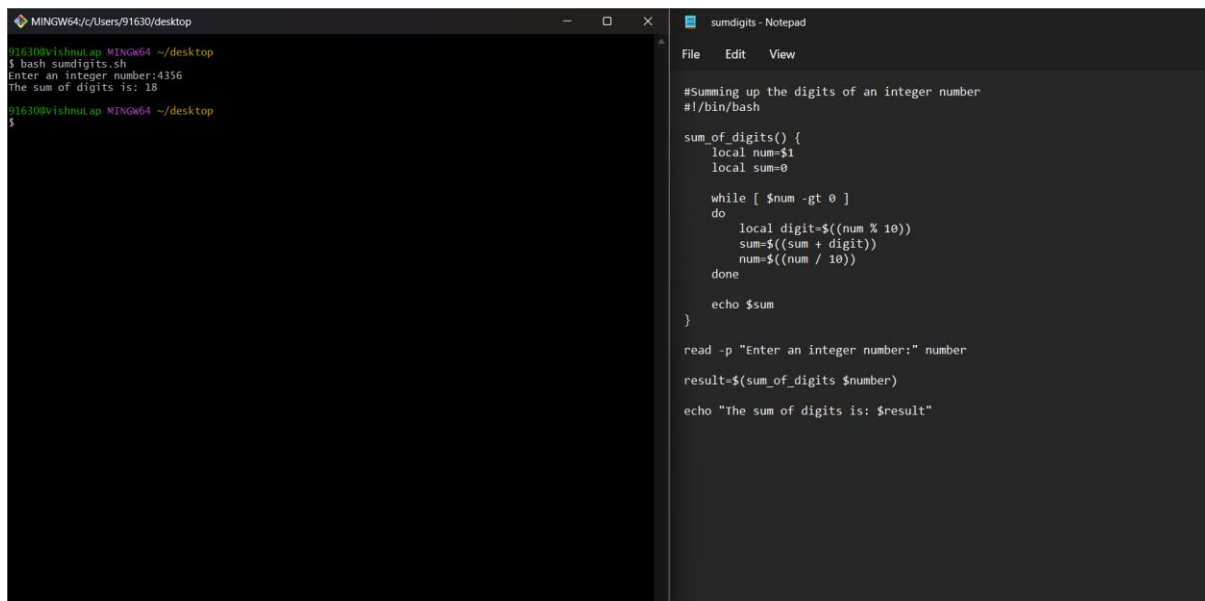
```
#Printing digits of an integer number
#!/bin/bash

print_digits() {
    num=$1
    num_str="$num"

    for ((i = 0; i < ${#num_str}; i++))
    do
        digit="${num_str:i:1}"
        echo "$digit"
    done
}

read -p "Enter an integer: " number
print_digits $number
```

## Exp-18



The screenshot shows a terminal window on the left and a Notepad window titled 'sumdigits - Notepad' on the right. The terminal shows the execution of a script named 'sumdigits.sh'. The user enters '4356' and the script outputs 'The sum of digits is: 18'. The Notepad window contains the following code:

```
#Summing up the digits of an integer number
#!/bin/bash

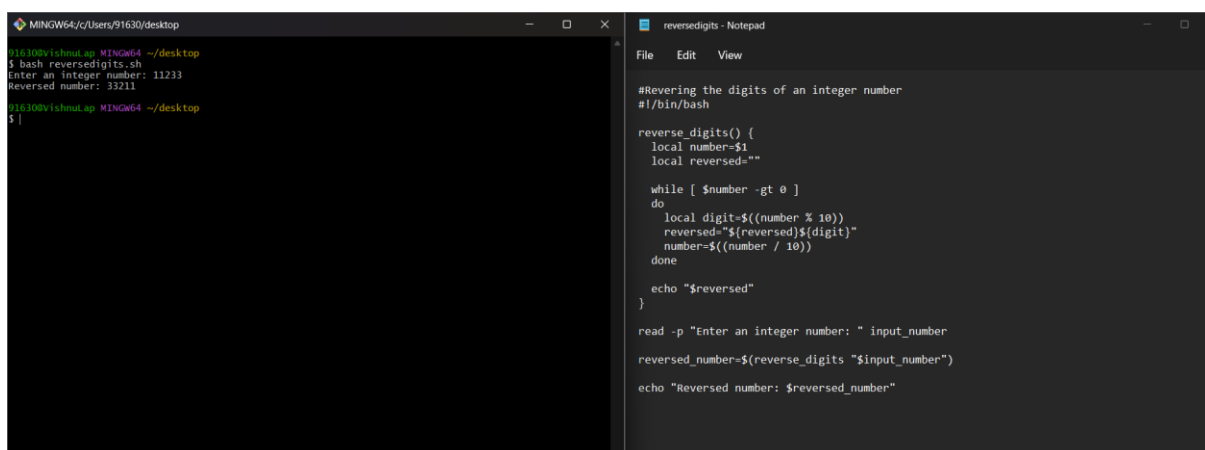
sum_of_digits() {
    local num=$1
    local sum=0

    while [ $num -gt 0 ]
    do
        local digit=$((num % 10))
        sum=$((sum + digit))
        num=$((num / 10))
    done

    echo $sum
}

read -p "Enter an integer number:" number
result=$(sum_of_digits $number)
echo "The sum of digits is: $result"
```

## Exp-19



The screenshot shows a terminal window on the left and a Notepad window titled 'reversedigits - Notepad' on the right. The terminal shows the execution of a script named 'reversedigits.sh'. The user enters '11233' and the script outputs 'Reversed number: 33211'. The Notepad window contains the following code:

```
#Revering the digits of an integer number
#!/bin/bash

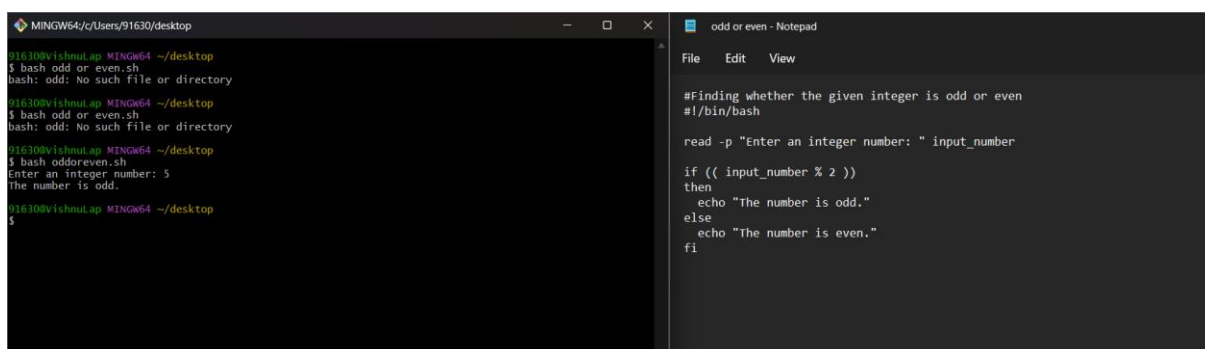
reverse_digits() {
    local number=$1
    local reversed=""

    while [ $number -gt 0 ]
    do
        local digit=$((number % 10))
        reversed="$digit${reversed}"
        number=$((number / 10))
    done

    echo "$reversed"
}

read -p "Enter an integer number: " input_number
reversed_number=$(reverse_digits "$input_number")
echo "Reversed number: $reversed_number"
```

## Exp-20



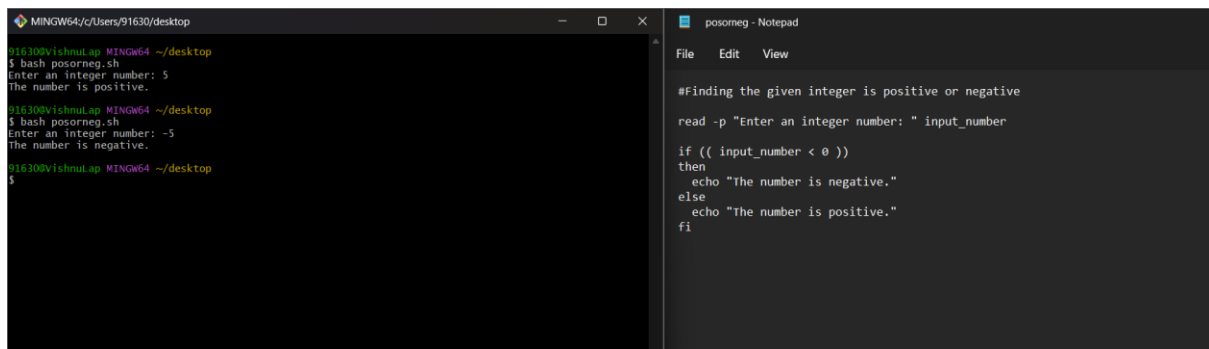
The screenshot shows a terminal window on the left and a Notepad window titled 'odd or even - Notepad' on the right. The terminal shows the execution of a script named 'oddeven.sh'. The user enters '5' and the script outputs 'The number is odd.'. The Notepad window contains the following code:

```
#Finding whether the given integer is odd or even
#!/bin/bash

read -p "Enter an integer number: " input_number

if (( input_number % 2 ))
then
    echo "The number is odd."
else
    echo "The number is even."
fi
```

## Exp-21



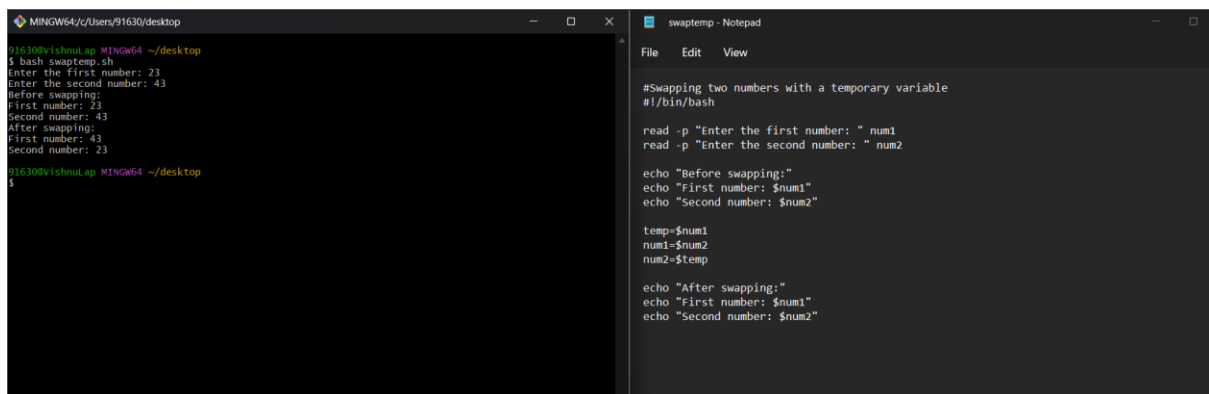
The screenshot shows a Windows terminal window titled "MINGW64/c/Users/91630/desktop" and a Notepad window titled "posorneg - Notepad". The terminal displays the execution of a shell script named "posorneg.sh". The user enters the number 5, and the script outputs "The number is positive." The user then enters -5, and the script outputs "The number is negative." The Notepad window shows the source code of the script.

```
#Finding the given integer is positive or negative

read -p "Enter an integer number: " input_number

if (( input_number < 0 ))
then
    echo "The number is negative."
else
    echo "The number is positive."
fi
```

## Exp-22



The screenshot shows a Windows terminal window titled "MINGW64/c/Users/91630/desktop" and a Notepad window titled "swaptemp - Notepad". The terminal displays the execution of a shell script named "swaptemp.sh". The user enters the first number 23 and the second number 43. The script outputs "Before swapping: First number: 23, Second number: 43" and "After swapping: First number: 43, Second number: 23". The Notepad window shows the source code of the script.

```
#Swapping two numbers with a temporary variable
#!/bin/bash

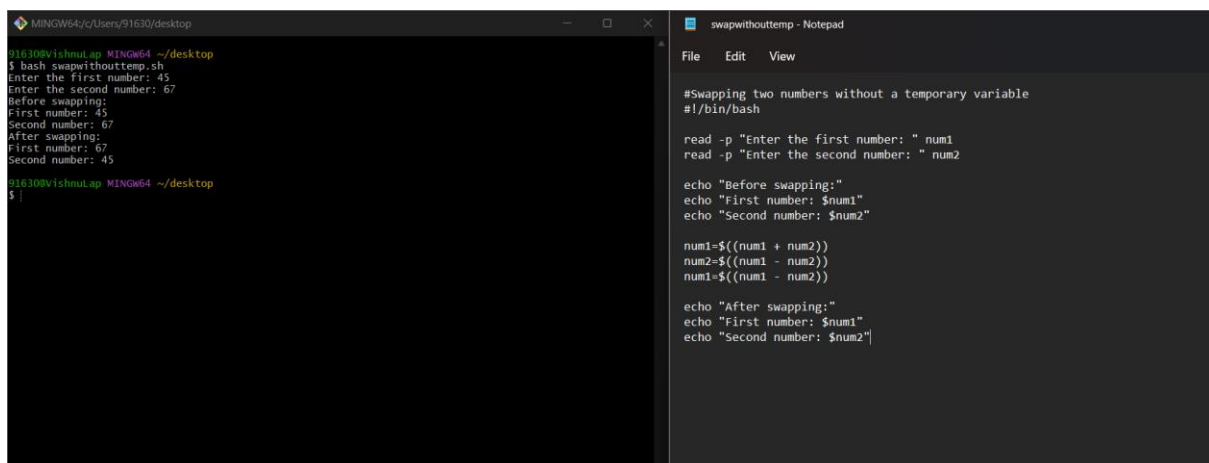
read -p "Enter the first number: " num1
read -p "Enter the second number: " num2

echo "Before swapping:"
echo "First number: $num1"
echo "Second number: $num2"

temp=$num1
num1=$num2
num2=$temp

echo "After swapping:"
echo "First number: $num1"
echo "Second number: $num2"
```

## Exp-23



The screenshot shows a Windows terminal window titled "MINGW64/c/Users/91630/desktop" and a Notepad window titled "swapwithouttemp - Notepad". The terminal displays the execution of a shell script named "swapwithouttemp.sh". The user enters the first number 45 and the second number 67. The script outputs "Before swapping: First number: 45, Second number: 67" and "After swapping: First number: 67, Second number: 45". The Notepad window shows the source code of the script.

```
#Swapping two numbers without a temporary variable
#!/bin/bash

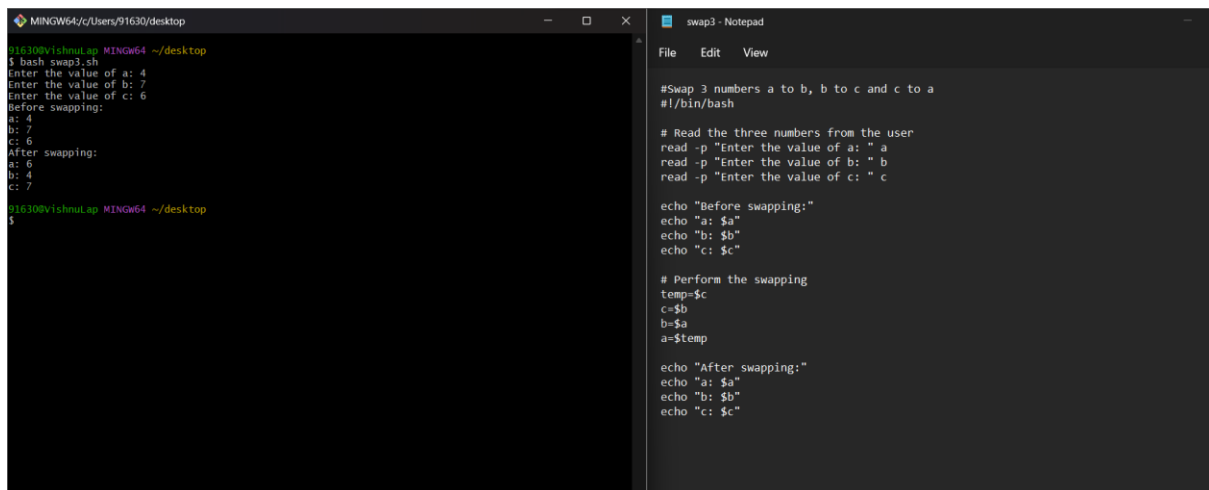
read -p "Enter the first number: " num1
read -p "Enter the second number: " num2

echo "Before swapping:"
echo "First number: $num1"
echo "Second number: $num2"

num1=$((num1 + num2))
num2=$((num1 - num2))
num1=$((num1 - num2))

echo "After swapping:"
echo "First number: $num1"
echo "Second number: $num2"
```

## Exp-24



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'swap3 - Notepad'. The terminal displays the execution of a script named 'swap3.sh'. The script prompts the user to enter three numbers: a, b, and c. It then displays the values before and after swapping them. The Notepad window shows the source code of the script.

```
#Swap 3 numbers a to b, b to c and c to a
#!/bin/bash

# Read the three numbers from the user
read -p "Enter the value of a: " a
read -p "Enter the value of b: " b
read -p "Enter the value of c: " c

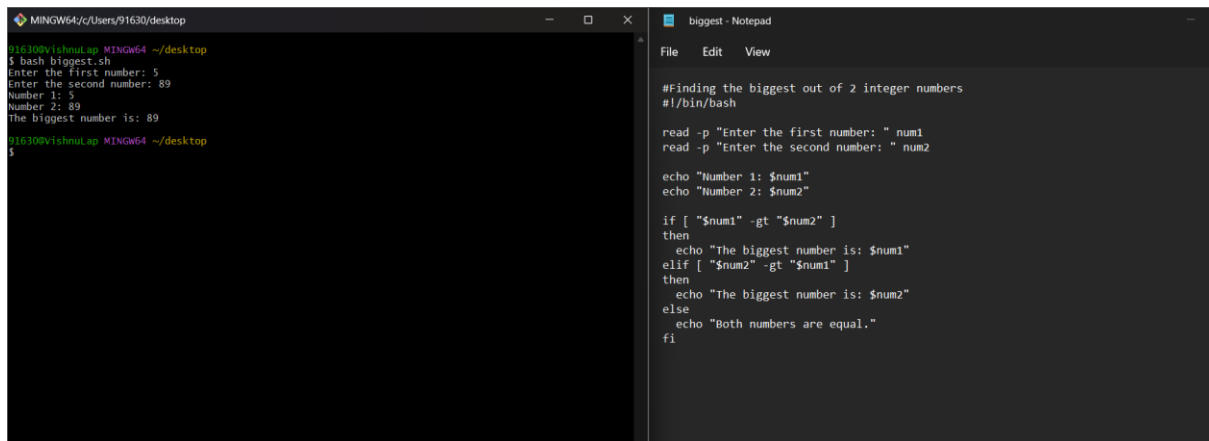
echo "Before swapping:"
echo "a: $a"
echo "b: $b"
echo "c: $c"

# Perform the swapping
temp=$c
c=$b
b=$a
a=$temp

echo "After swapping:"
echo "a: $a"
echo "b: $b"
echo "c: $c"
```

```
91630@vishnu.ap MINGW64 ~/desktop
$ bash swap3.sh
Enter the value of a: 4
Enter the value of b: 7
Enter the value of c: 6
Before swapping:
a: 4
b: 7
c: 6
After swapping:
a: 6
b: 4
c: 7
91630@vishnu.ap MINGW64 ~/desktop
$
```

## Exp-25



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'biggest - Notepad'. The terminal displays the execution of a script named 'biggest.sh'. The script prompts the user to enter two numbers, num1 and num2, and then displays the biggest number or a message if they are equal. The Notepad window shows the source code of the script.

```
#Finding the biggest out of 2 integer numbers
#!/bin/bash

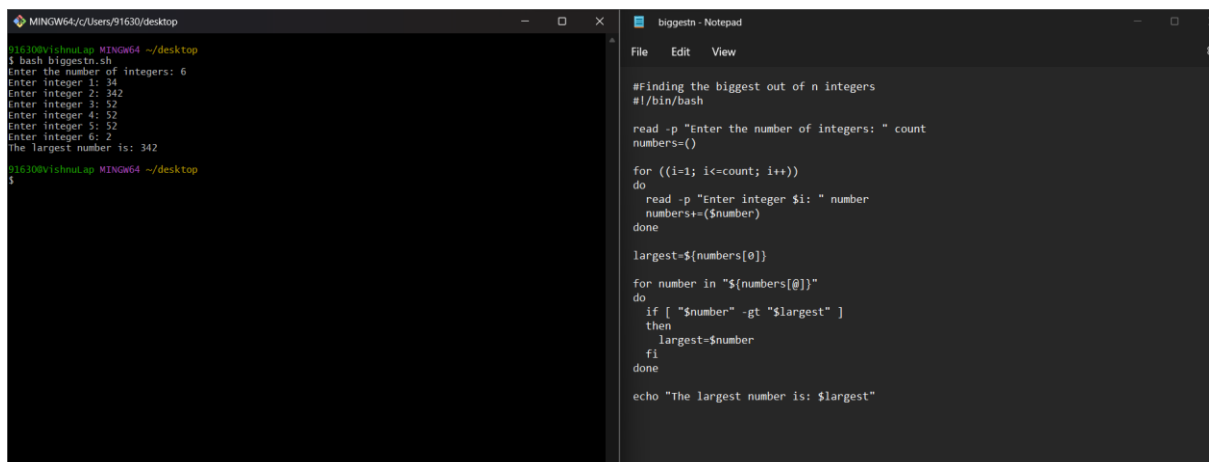
read -p "Enter the first number: " num1
read -p "Enter the second number: " num2

echo "Number 1: $num1"
echo "Number 2: $num2"

if [ "$num1" -gt "$num2" ]
then
    echo "The biggest number is: $num1"
elif [ "$num2" -gt "$num1" ]
then
    echo "The biggest number is: $num2"
else
    echo "Both numbers are equal."
fi
```

```
91630@vishnu.ap MINGW64 ~/desktop
$ bash biggest.sh
Enter the first number: 5
Enter the second number: 89
Number 1: 5
Number 2: 89
The biggest number is: 89
91630@vishnu.ap MINGW64 ~/desktop
$
```

## Exp-26



The screenshot shows a Windows terminal window titled 'MINGW64/c/Users/91630/desktop' and a Notepad window titled 'biggestn - Notepad'. The terminal displays the execution of a script named 'biggestn.sh'. The script prompts the user to enter the number of integers, count, and then enters each integer. It then displays the largest number. The Notepad window shows the source code of the script.

```
#Finding the biggest out of n integers
#!/bin/bash

read -p "Enter the number of integers: " count
numbers=()

for ((i=1; i<=count; i++))
do
    read -p "Enter integer $i: " number
    numbers+=($number)
done

largest=${numbers[0]}

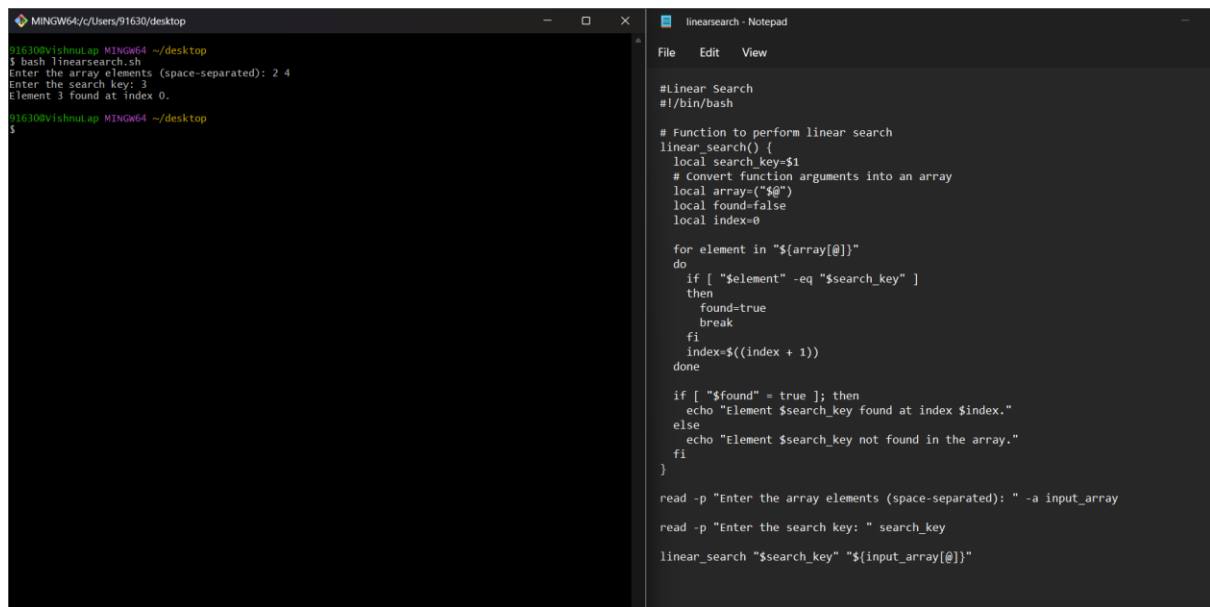
for number in "${numbers[@]}"
do
    if [ "$number" -gt "$largest" ]
    then
        largest=$number
    fi
done

echo "The largest number is: $largest"
```

```
91630@vishnu.ap MINGW64 ~/desktop
$ bash biggestn.sh
Enter the number of integers: 6
Enter integer 1: 34
Enter integer 2: 342
Enter integer 3: 52
Enter integer 4: 52
Enter integer 5: 52
Enter integer 6: 2
The largest number is: 342
91630@vishnu.ap MINGW64 ~/desktop
$
```



## Exp-30



The image shows a screenshot of a computer screen with two windows. The left window is a terminal titled 'MINGW64/C:/Users/91630/desktop' showing the execution of a script named 'linearsearch.sh'. The user enters '2 4' for array elements and '3' for the search key. The script outputs 'Element 3 found at index 0.' The right window is a Notepad file titled 'linearsearch - Notepad' containing the source code of the script.

```
#Linear Search
#!/bin/bash

# Function to perform linear search
linear_search() {
    local search_key=$1
    # Convert function arguments into an array
    local array=("$@")
    local found=false
    local index=0

    for element in "${array[@]}"
    do
        if [ "$element" -eq "$search_key" ]
        then
            found=true
            break
        fi
        index=$((index + 1))
    done

    if [ "$found" = true ]; then
        echo "Element $search_key found at index $index."
    else
        echo "Element $search_key not found in the array."
    fi
}

read -p "Enter the array elements (space-separated): " -a input_array
read -p "Enter the search key: " search_key
linear_search "$search_key" "${input_array[@]}"
```

```
91630@vishnut.ap MINGW64 ~/desktop
$ bash linearsearch.sh
Enter the array elements (space-separated): 2 4
Enter the search key: 3
Element 3 found at index 0.
91630@vishnut.ap MINGW64 ~/desktop
$
```