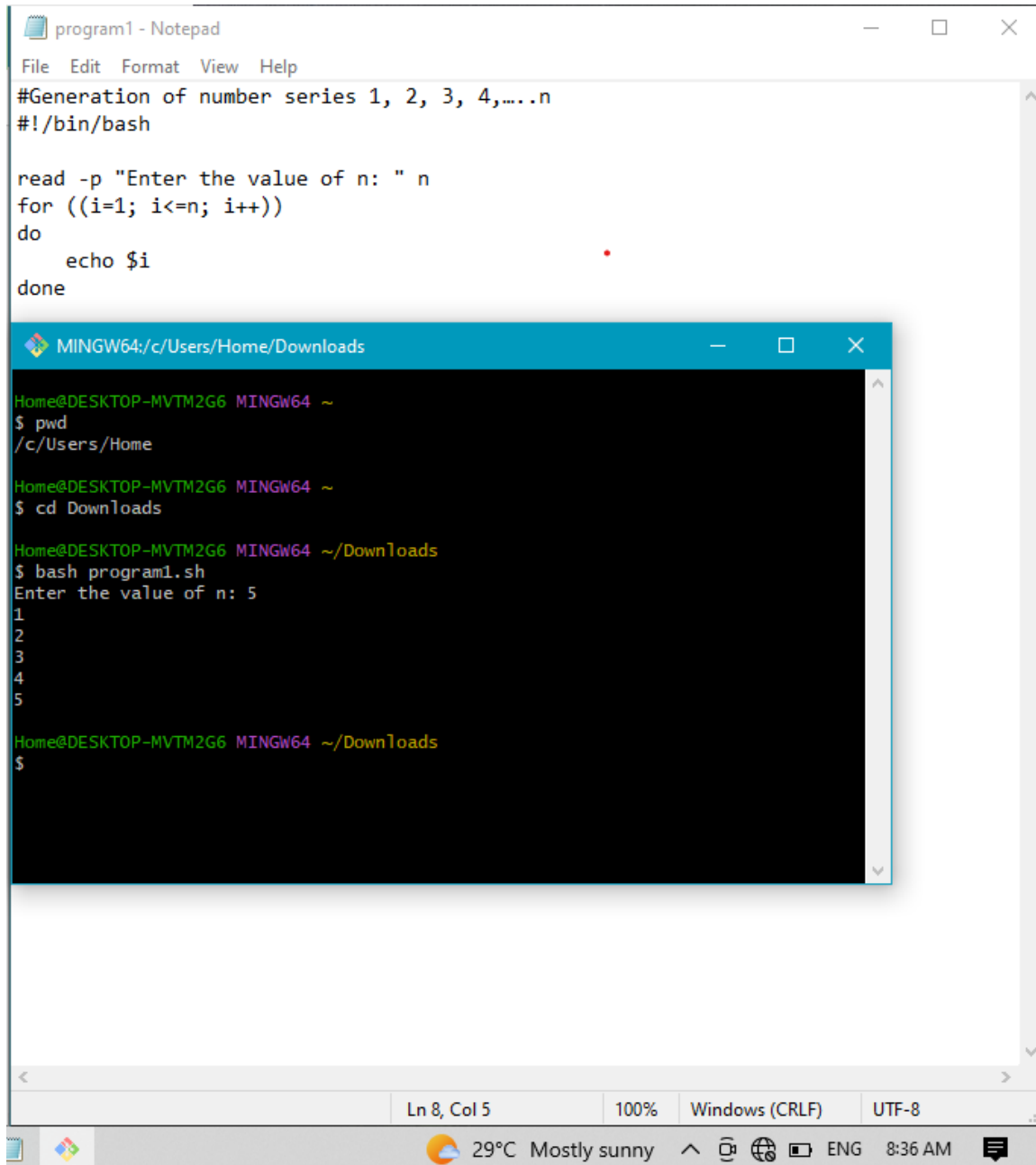


SHELL PROGRAMS

PROGRAM 1

CODE:



The image shows a Notepad window titled 'program1 - Notepad' with a menu bar (File, Edit, Format, View, Help). The script content is as follows:

```
#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
```

Overlaid on the Notepad window is a terminal window titled 'MINGW64:/c/Users/Home/Downloads'. The terminal shows the following session:

```
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

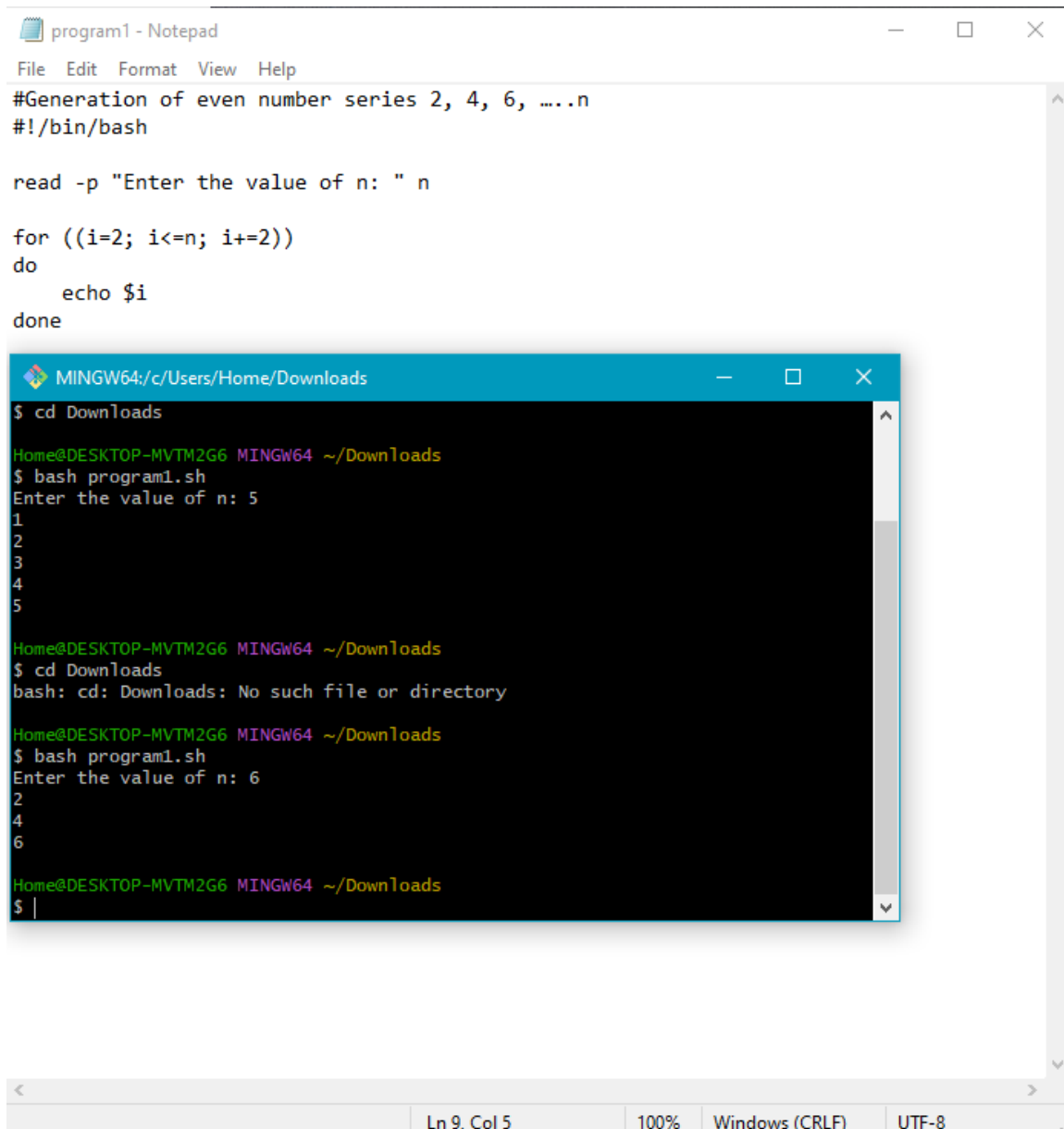
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program1.sh
Enter the value of n: 5
1
2
3
4
5

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

The Notepad window's status bar at the bottom indicates 'Ln 8, Col 5', '100%', 'Windows (CRLF)', and 'UTF-8'. The system taskbar at the very bottom shows the date and time as '8:36 AM' along with other system icons.

program 2:

CODE:



The image shows a Notepad window titled 'program1 - Notepad' containing a shell script. Below it is a terminal window titled 'MINGW64; c:/Users/Home/Downloads' showing the script being executed. The script generates an even number series based on user input 'n'.

```
program1 - Notepad
File Edit Format View Help
#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
```

Terminal Output:

```
MINGW64; c:/Users/Home/Downloads
$ cd Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program1.sh
Enter the value of n: 5
1
2
3
4
5

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ cd Downloads
bash: cd: Downloads: No such file or directory

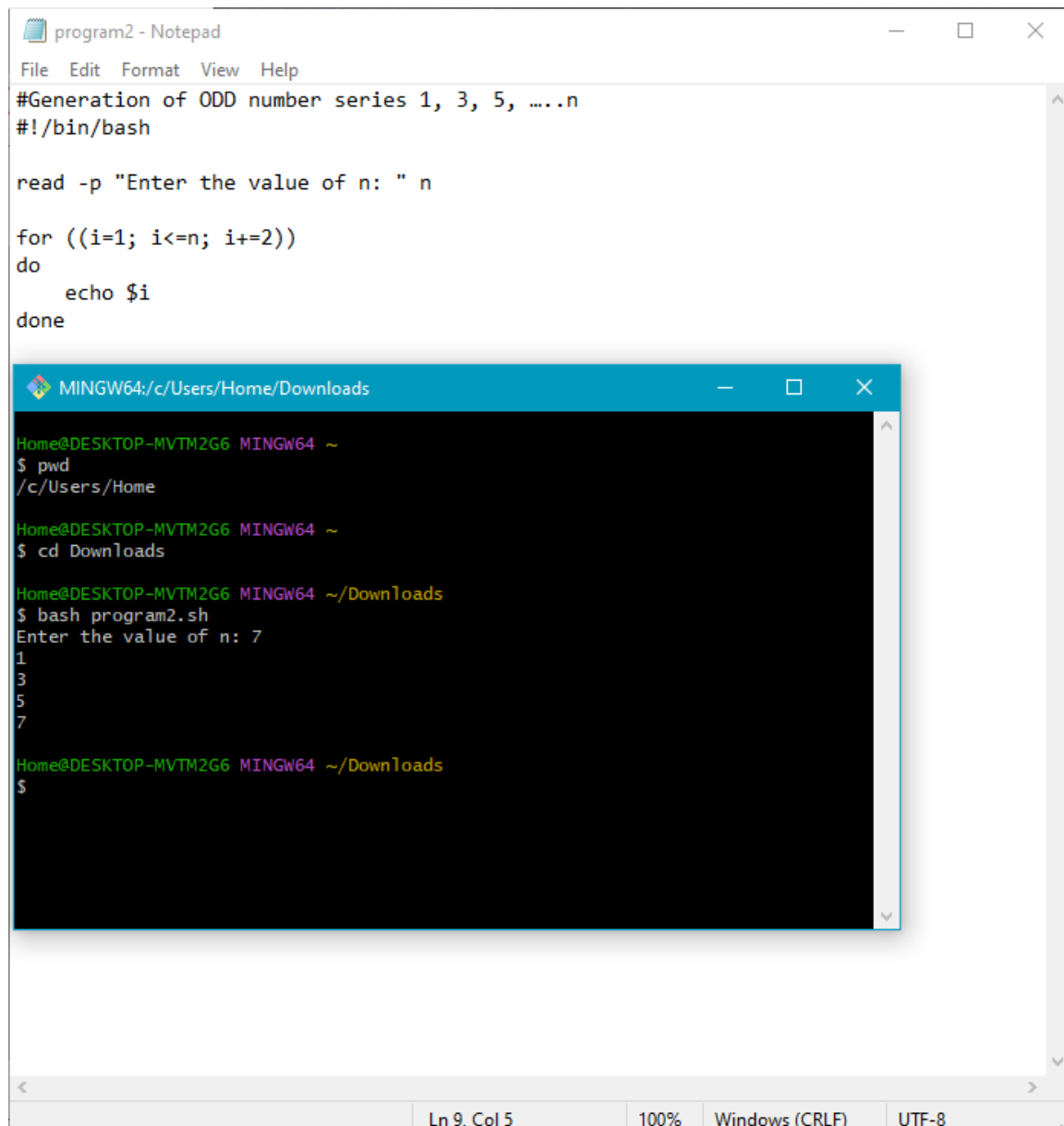
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program1.sh
Enter the value of n: 6
2
4
6

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ |
```

Notepad status bar: Ln 9, Col 5 | 100% | Windows (CRLF) | UTF-8

PROGRAM 3:

CODE:



The image shows a Notepad window titled "program2 - Notepad" containing a shell script. The script generates an odd number series. Below the Notepad window, a terminal window titled "MINGW64:/c/Users/Home/Downloads" shows the execution of the script. The user enters '7' for 'n', and the script outputs the series: 1, 3, 5, 7.

```
program2 - Notepad
File Edit Format View Help
#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
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

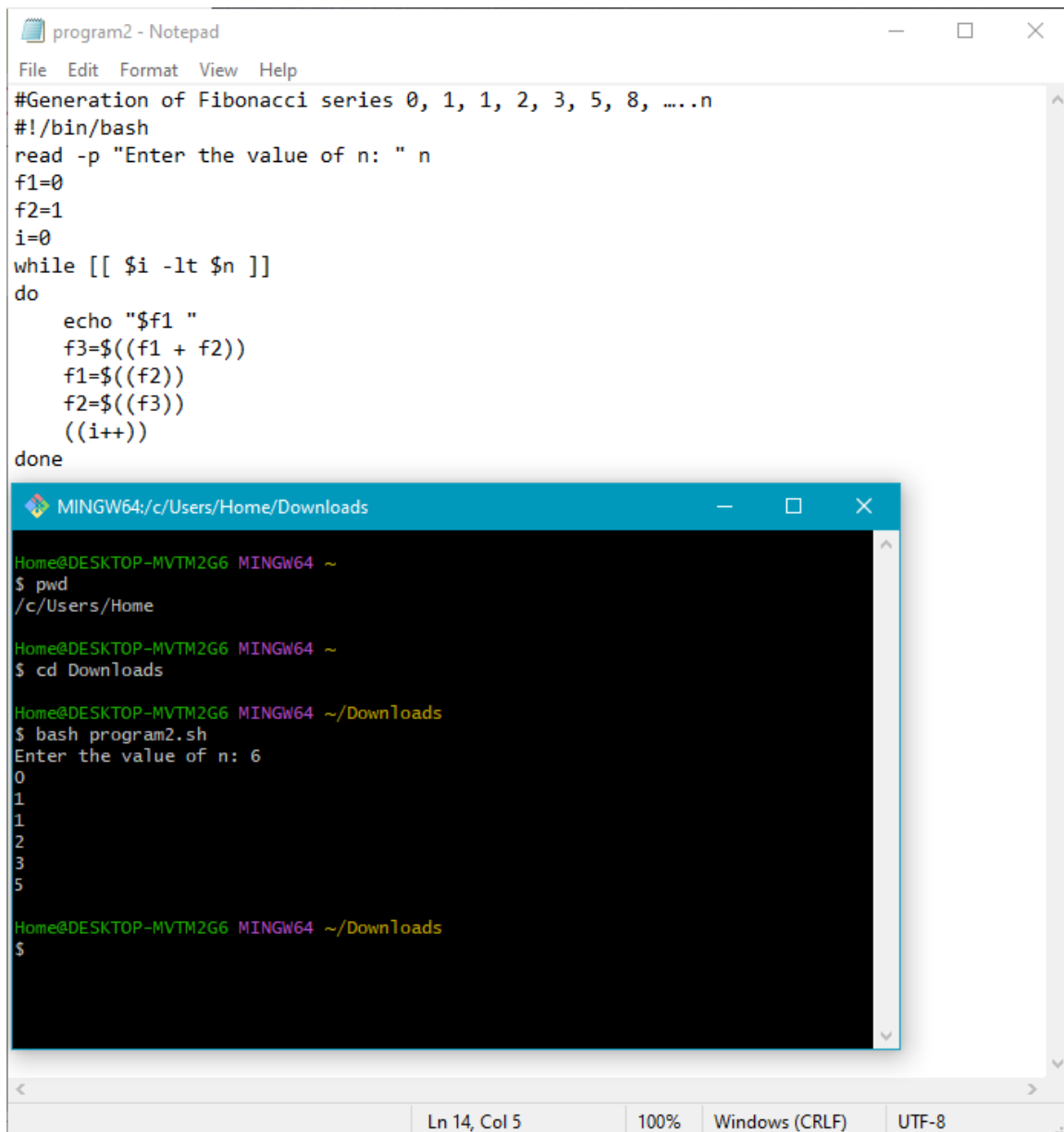
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program2.sh
Enter the value of n: 7
1
3
5
7

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 9, Col 5 | 100% | Windows (CRLF) | UTF-8

PROGRAM 4:

CODE:



The image shows a Notepad window titled "program2 - Notepad" containing a shell script for generating a Fibonacci series. The script prompts the user for a value 'n' and prints the series up to that value. Below the Notepad window is a terminal window titled "MINGW64:/c/Users/Home/Downloads" showing the execution of the script. The user enters '6', and the terminal outputs the Fibonacci series: 0, 1, 1, 2, 3, 5.

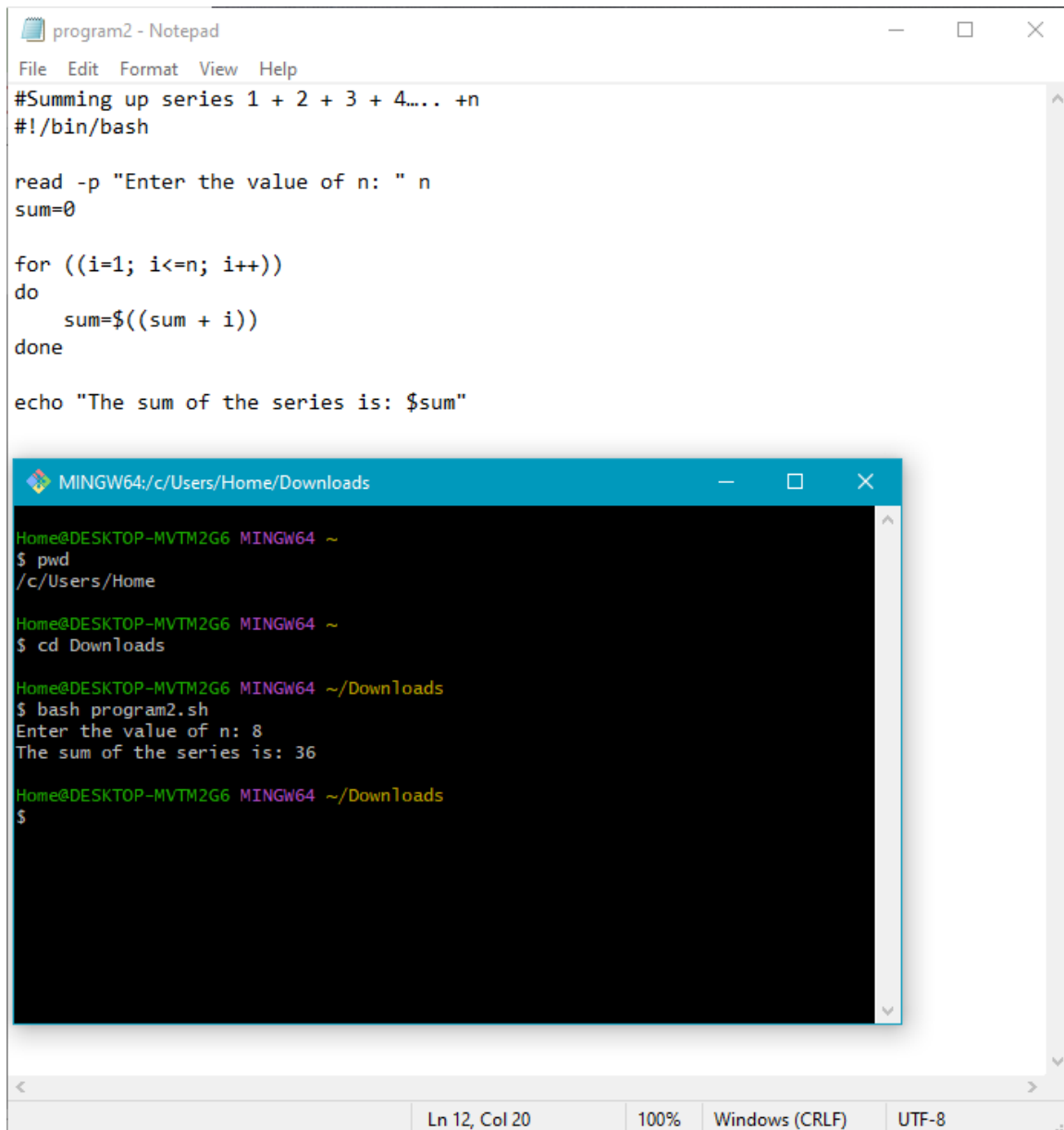
```
program2 - Notepad
File Edit Format View Help
#Generation of Fibonacci series 0, 1, 1, 2, 3, 5, ....n
#!/bin/bash
read -p "Enter the value of n: " n
f1=0
f2=1
i=0
while [[ $i -lt $n ]]
do
    echo "$f1 "
    f3=$((f1 + f2))
    f1=$((f2))
    f2=$((f3))
    ((i++))
done

MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program2.sh
Enter the value of n: 6
0
1
1
2
3
5
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 14, Col 5 100% Windows (CRLF) UTF-8

PROGRAM 5:

CODE:



The image shows two overlapping windows. The top window is a Notepad application titled 'program2 - Notepad'. It contains a shell script for summing a series. The bottom window is a terminal application titled 'MINGW64:/c/Users/Home/Downloads'. It shows the execution of the script, where the user enters '8' and the output is 'The sum of the series is: 36'.

```
program2 - Notepad
File Edit Format View Help
#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

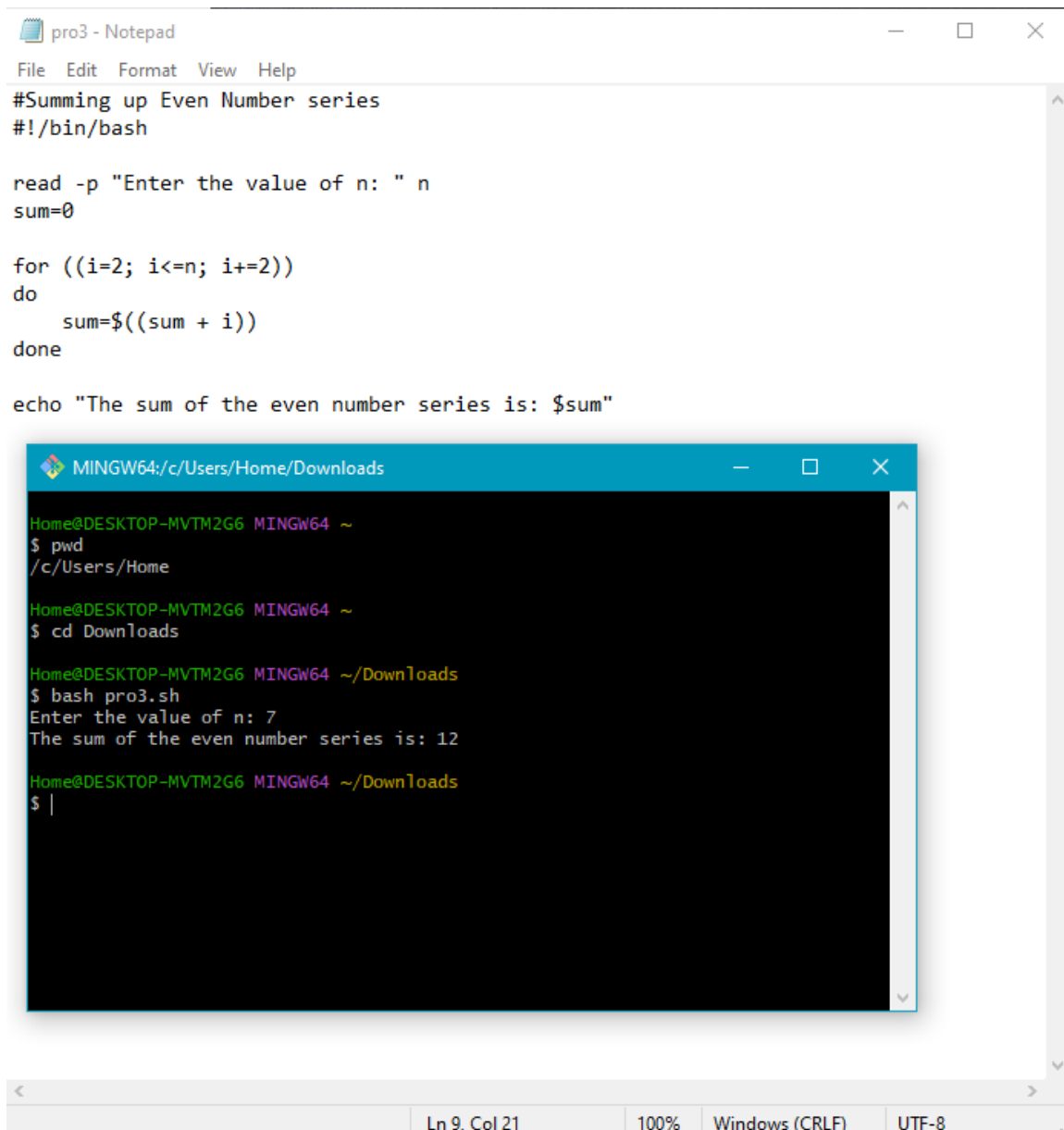
echo "The sum of the series is: $sum"
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash program2.sh
Enter the value of n: 8
The sum of the series is: 36
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 12, Col 20 100% Windows (CRLF) UTF-8

PROGRAM 6:

CODE:



The image shows a Notepad window titled 'pro3 - Notepad' with a menu bar (File, Edit, Format, View, Help). The text inside is a shell script for summing even numbers. Below the Notepad window is a terminal window titled 'MINGW64:/c/Users/Home/Downloads'. The terminal shows the execution of the script, where the user enters '7' and the output is 'The sum of the even number series is: 12'.

```
pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

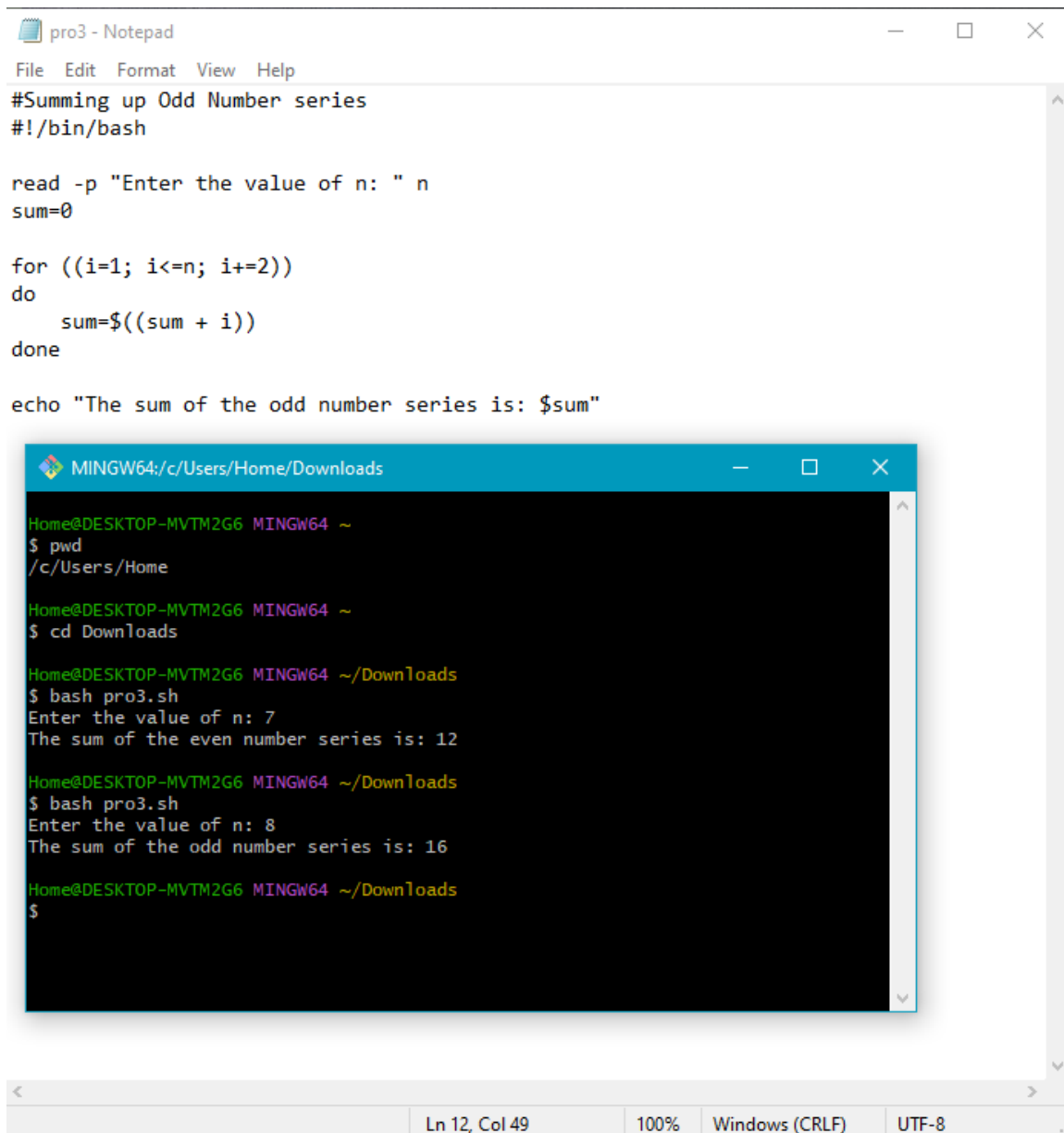
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 7
The sum of the even number series is: 12

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ |
```

Ln 9, Col 21 | 100% | Windows (CRLF) | UTF-8

PROGRAM 7:

CODE :



The image shows a Notepad window titled 'pro3 - Notepad' with a menu bar (File, Edit, Format, View, Help). The script content is as follows:

```
#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"
```

Below the Notepad window is a terminal window titled 'MINGW64:/c/Users/Home/Downloads'. It shows the following commands and output:

```
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 7
The sum of the even number series is: 12

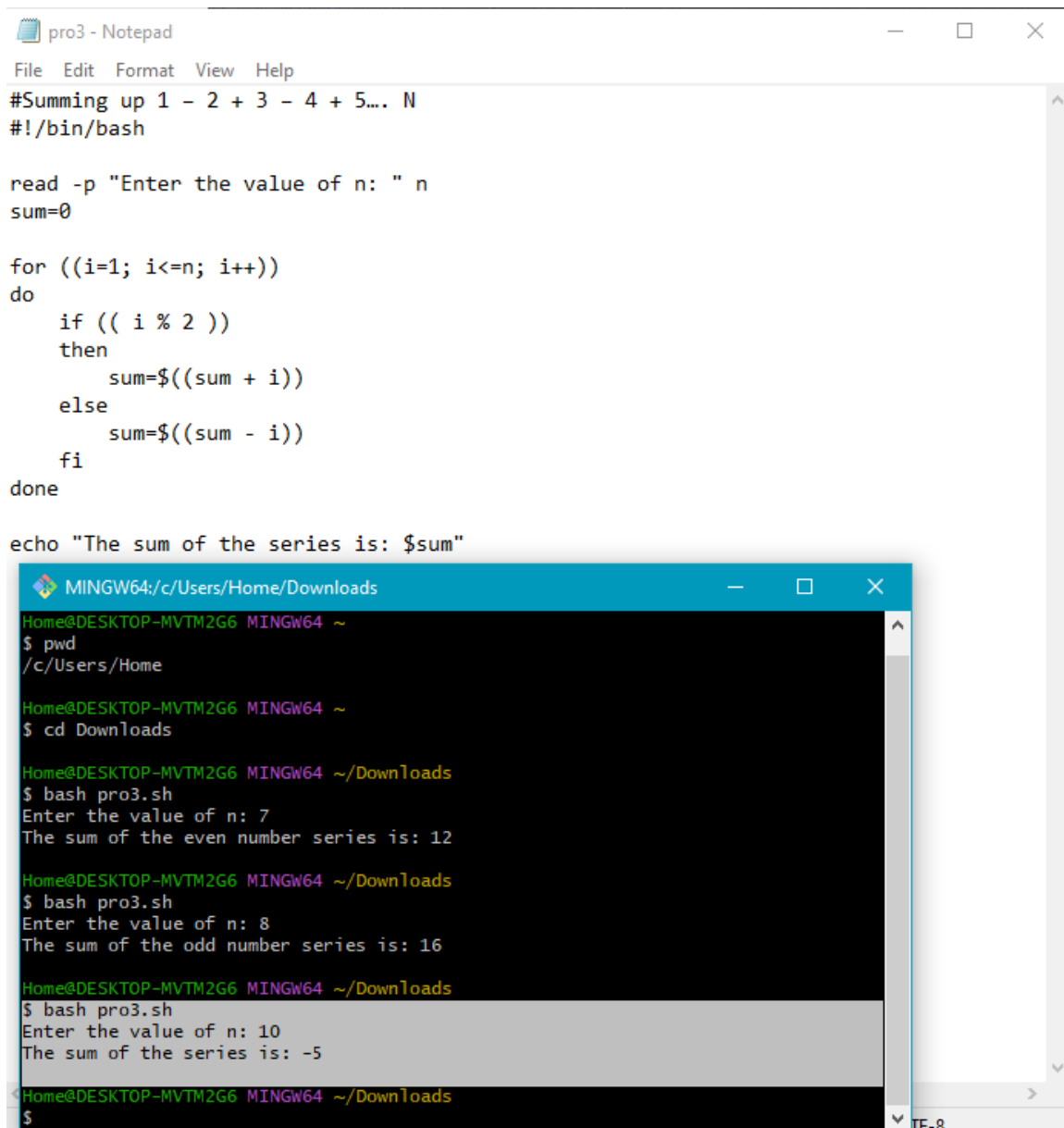
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 8
The sum of the odd number series is: 16

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

The status bar at the bottom of the Notepad window indicates 'Ln 12, Col 49', '100%', 'Windows (CRLF)', and 'UTF-8'.

PROGRAM 8:

CODE :



The image shows a Notepad window titled 'pro3 - Notepad' containing a shell script. Below it is a terminal window titled 'MINGW64:/c/Users/Home/Downloads' showing the script being executed with different inputs.

```
pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 7
The sum of the even number series is: 12

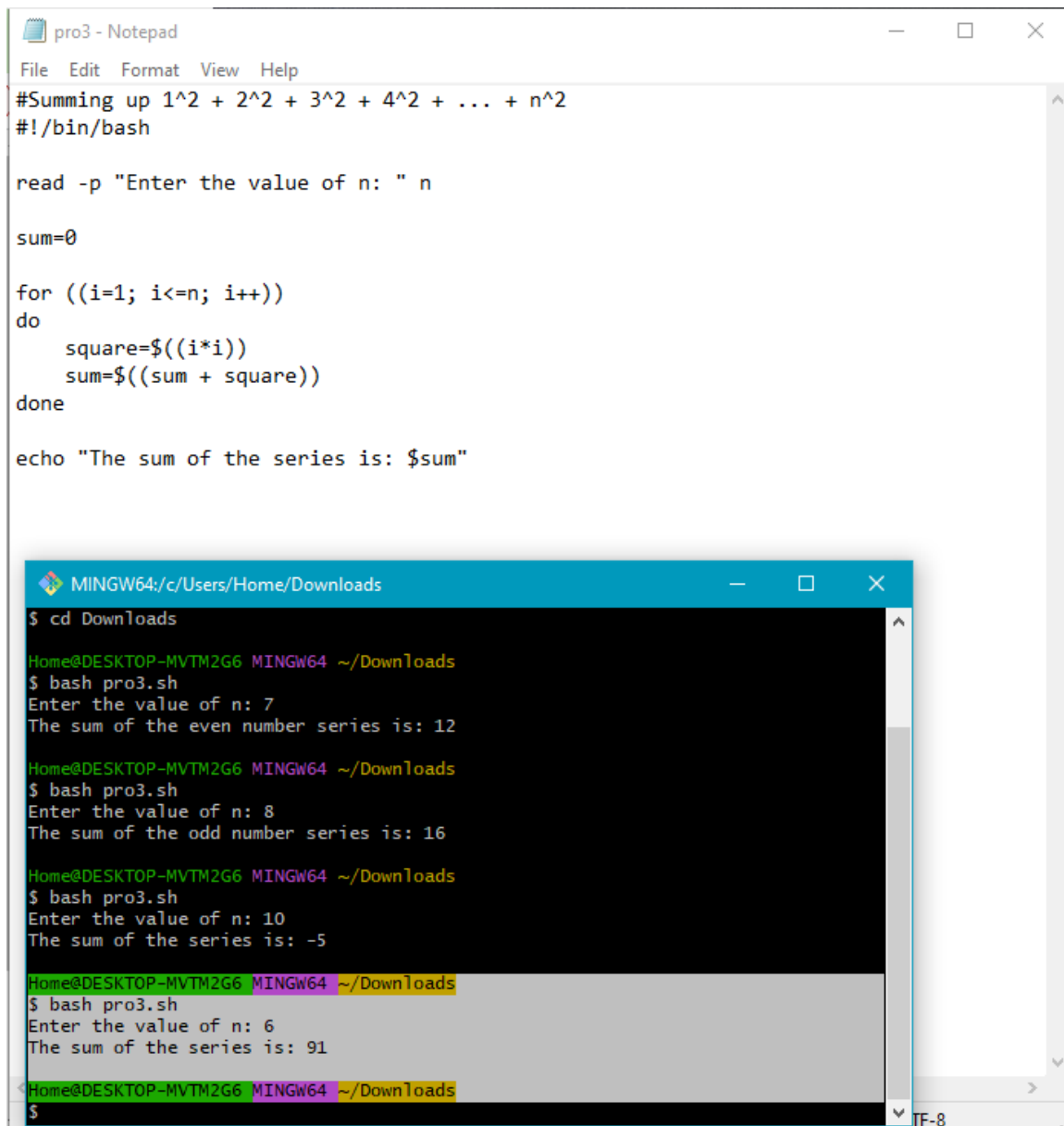
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 8
The sum of the odd number series is: 16

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 10
The sum of the series is: -5

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 9:

CODE:



The image shows two overlapping windows. The top window is a Notepad application titled 'pro3 - Notepad'. It contains a shell script for calculating the sum of squares from 1 to n. The script uses a for loop and arithmetic expansion. The bottom window is a terminal window titled 'MINGW64: c:/Users/Home/Downloads'. It shows the execution of the script 'pro3.sh' with different values of n (7, 8, 10, 6) and the resulting output.

```
pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64: c:/Users/Home/Downloads
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 7
The sum of the even number series is: 12

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 8
The sum of the odd number series is: 16

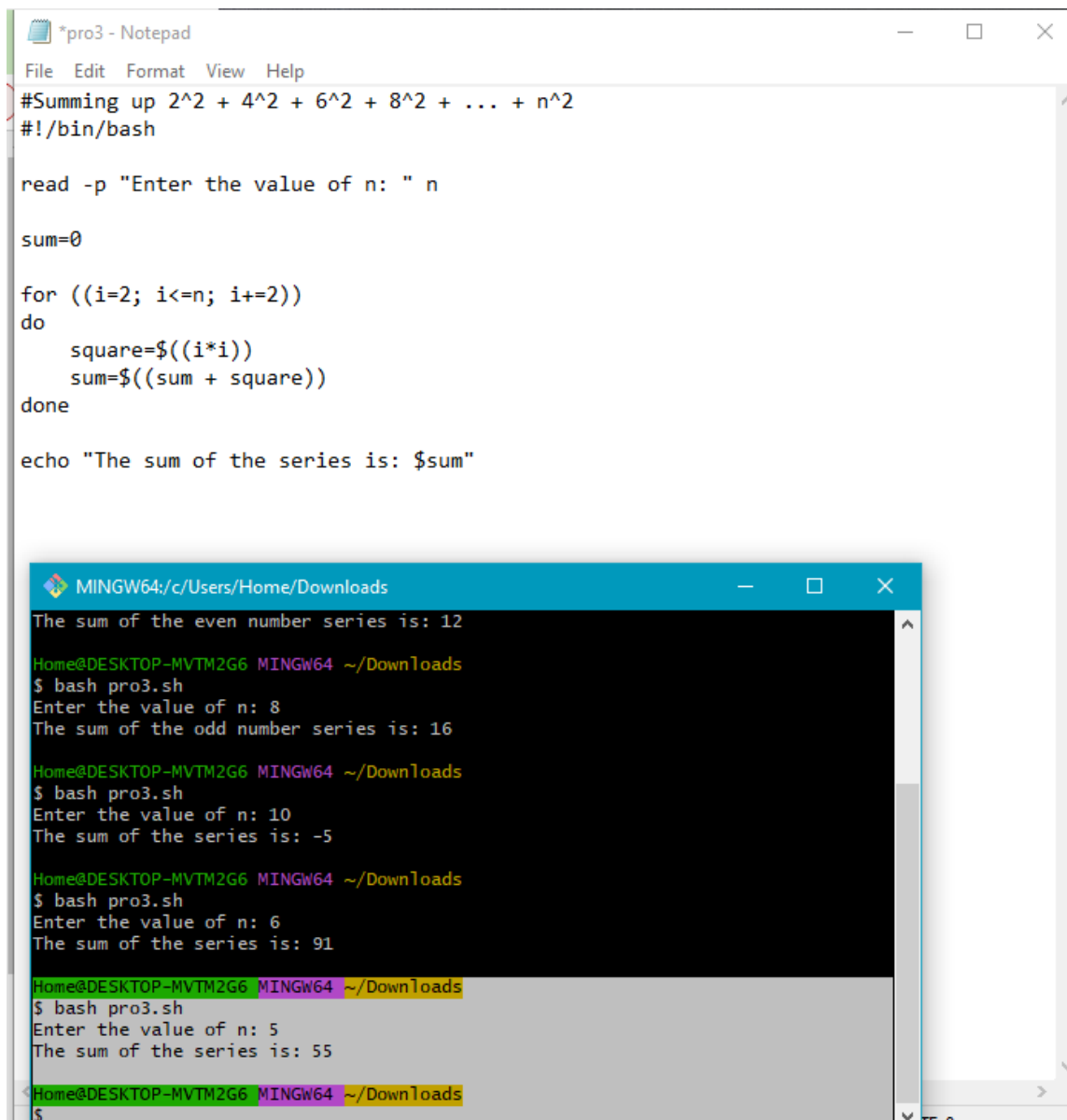
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 10
The sum of the series is: -5

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 6
The sum of the series is: 91

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 10:

CODE :



The image shows two windows. The top window is a Notepad editor titled '*pro3 - Notepad'. It contains a shell script for calculating the sum of squares of even numbers from 2 to n. The script uses a for loop and arithmetic expansion. The bottom window is a MINGW64 terminal titled 'MINGW64/c/Users/Home/Downloads'. It shows the execution of the script 'pro3.sh' with different values of n (8, 10, 6, 5) and the resulting output.

```
*pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64/c/Users/Home/Downloads
The sum of the even number series is: 12

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 8
The sum of the odd number series is: 16

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 10
The sum of the series is: -5

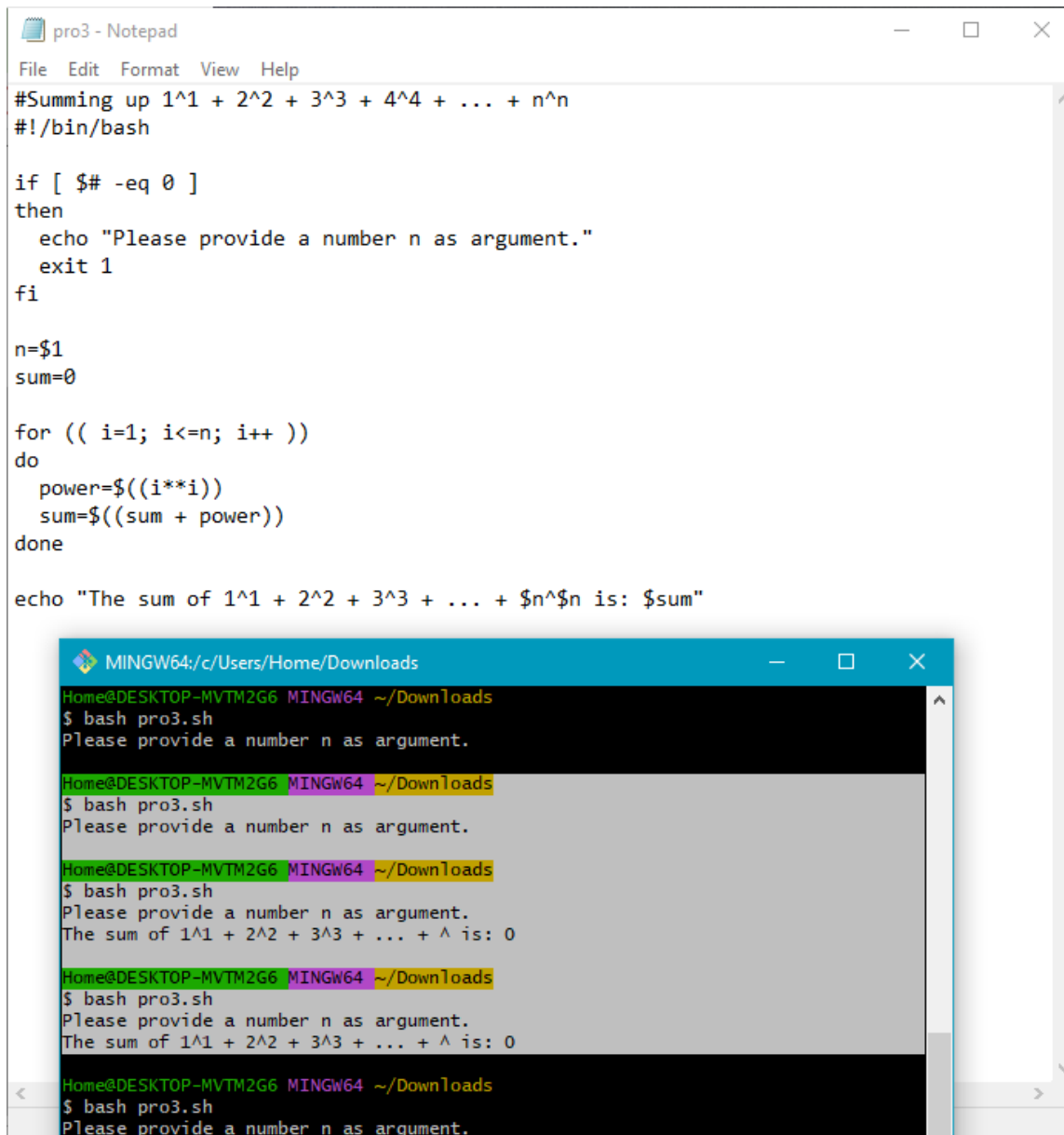
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 6
The sum of the series is: 91

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 5
The sum of the series is: 55

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 11:

CODE :



The image shows a Notepad window titled 'pro3 - Notepad' containing a shell script. The script calculates the sum of the first n natural numbers. Below the script, a terminal window titled 'MINGW64: c:/Users/Home/Downloads' shows the script being executed multiple times. The first two runs show the prompt 'Please provide a number n as argument.' The third run shows the output 'The sum of 1^1 + 2^2 + 3^3 + ... + ^ is: 0'. The fourth run shows the output 'The sum of 1^1 + 2^2 + 3^3 + ... + ^ is: 0'. The fifth run shows the prompt 'Please provide a number n as argument.'

```
pro3 - Notepad
File Edit Format View Help
#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

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"
```

MINGW64: c:/Users/Home/Downloads

```
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.

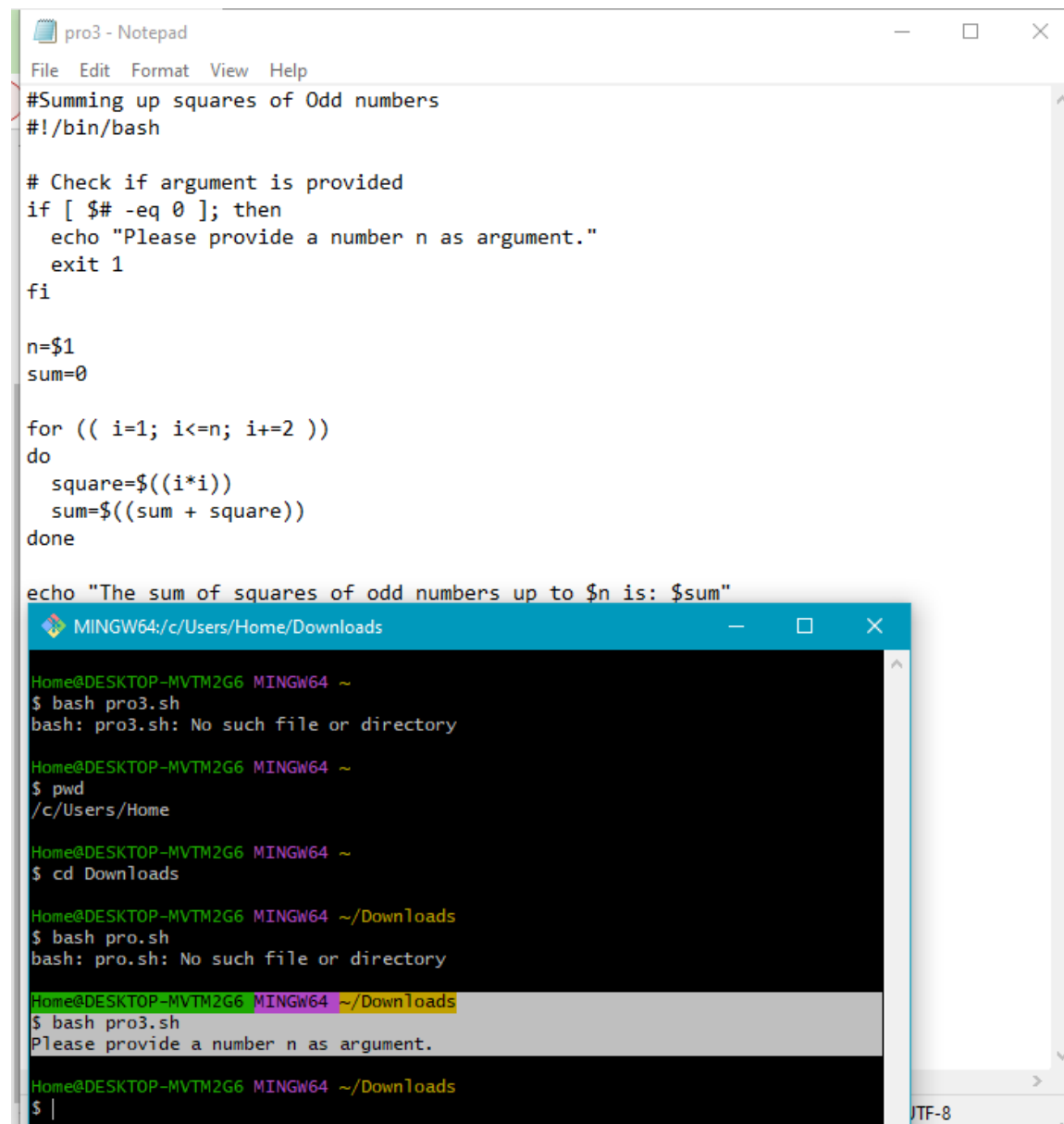
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.
The sum of 1^1 + 2^2 + 3^3 + ... + ^ is: 0

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.
The sum of 1^1 + 2^2 + 3^3 + ... + ^ is: 0

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.
```

PROGRAM 12:

CODE :



The image shows two overlapping windows. The top window is a Notepad application titled 'pro3 - Notepad'. It contains a shell script for summing the squares of odd numbers. The script starts with a comment '#Summing up squares of Odd numbers', followed by a shebang '#!/bin/bash'. It then checks if an argument is provided; if not, it prints 'Please provide a number n as argument.' and exits with status 1. If an argument is provided, it sets 'n=\$1' and 'sum=0'. A 'for' loop iterates from 'i=1' to 'i=n' in increments of 2. Inside the loop, it calculates 'square=\$((i*i))' and updates 'sum=\$((sum + square))'. After the loop, it prints 'The sum of squares of odd numbers up to \$n is: \$sum'.

The bottom window is a terminal titled 'MINGW64: c:/Users/Home/Downloads'. It shows the user attempting to run 'bash pro3.sh' from the home directory, which fails with 'bash: pro3.sh: No such file or directory'. After changing the directory to 'Downloads' with 'cd Downloads', the user runs 'bash pro3.sh' again. This time, it prompts 'Please provide a number n as argument.' and the user has entered a value, indicated by a cursor on the next line.

```
pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64: c:/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ bash pro3.sh
bash: pro3.sh: No such file or directory

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

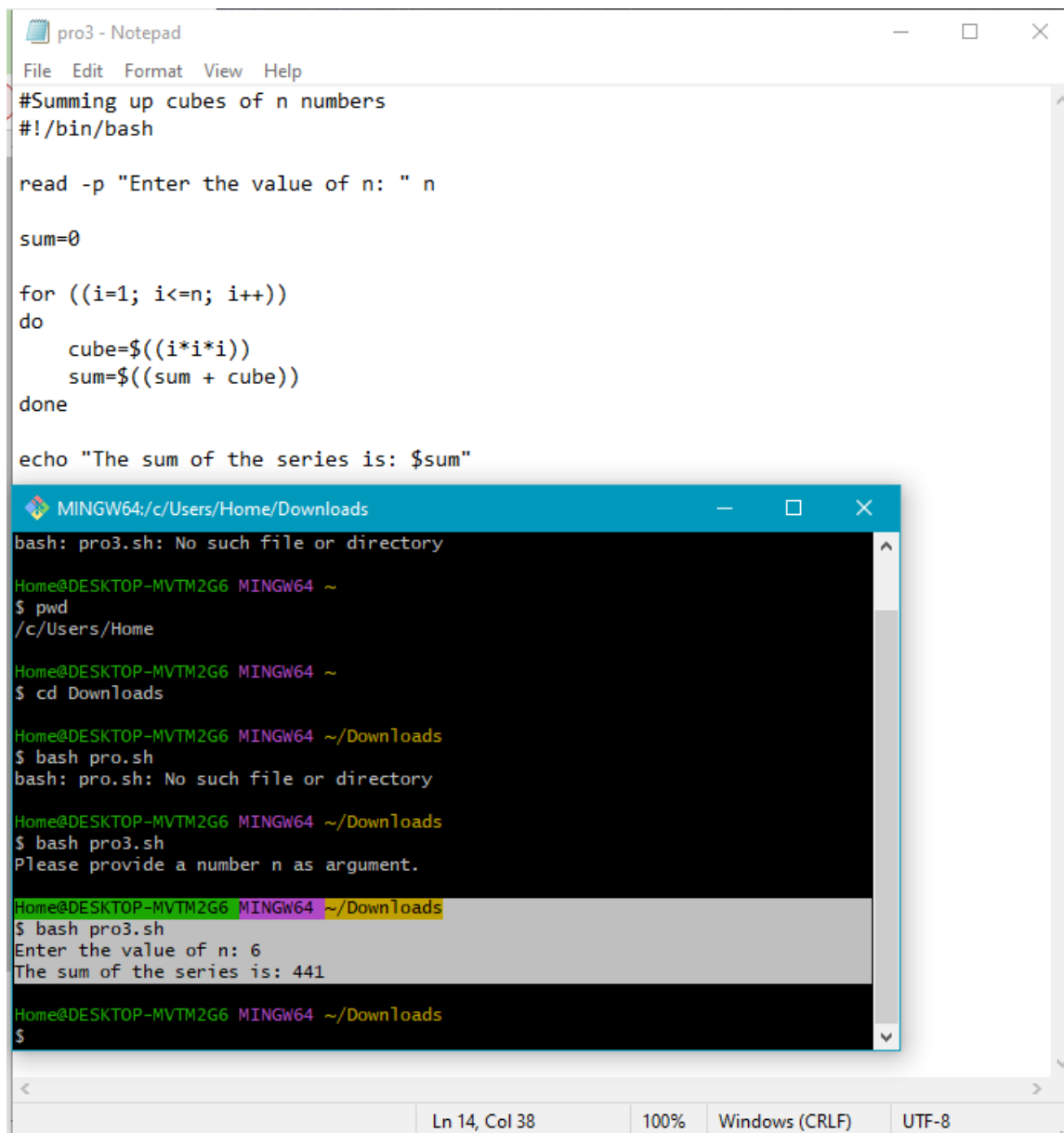
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
bash: pro3.sh: No such file or directory

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ |
```

PROGRAM 13:

CODE :



The image shows two overlapping windows. The top window is a Notepad application titled 'pro3 - Notepad'. It contains a shell script for calculating the sum of cubes of the first n natural numbers. The script uses a 'for' loop to iterate from 1 to n, calculate the cube of each number, and add it to a running sum. The bottom window is a terminal window titled 'MINGW64:/c/Users/Home/Downloads'. It shows the execution of the script. The user first runs 'bash pro3.sh' and receives an error. Then, they run 'bash pro3.sh' again, which prompts for a value of n. The user enters '6', and the script outputs 'The sum of the series is: 441'.

```
pro3 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64:/c/Users/Home/Downloads
bash: pro3.sh: No such file or directory

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
bash: pro3.sh: No such file or directory

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Please provide a number n as argument.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter the value of n: 6
The sum of the series is: 441

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 14, Col 38 100% Windows (CRLF) UTF-8

PROGRAM 14:

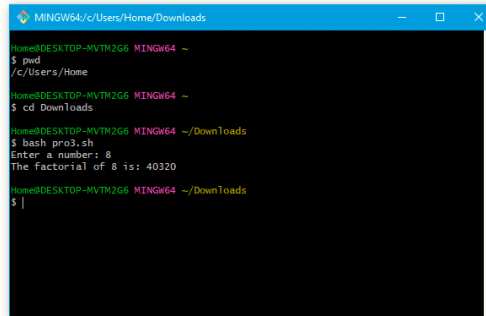
CODE :

```
pro3 - Notepad
File Edit Format View Help
#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

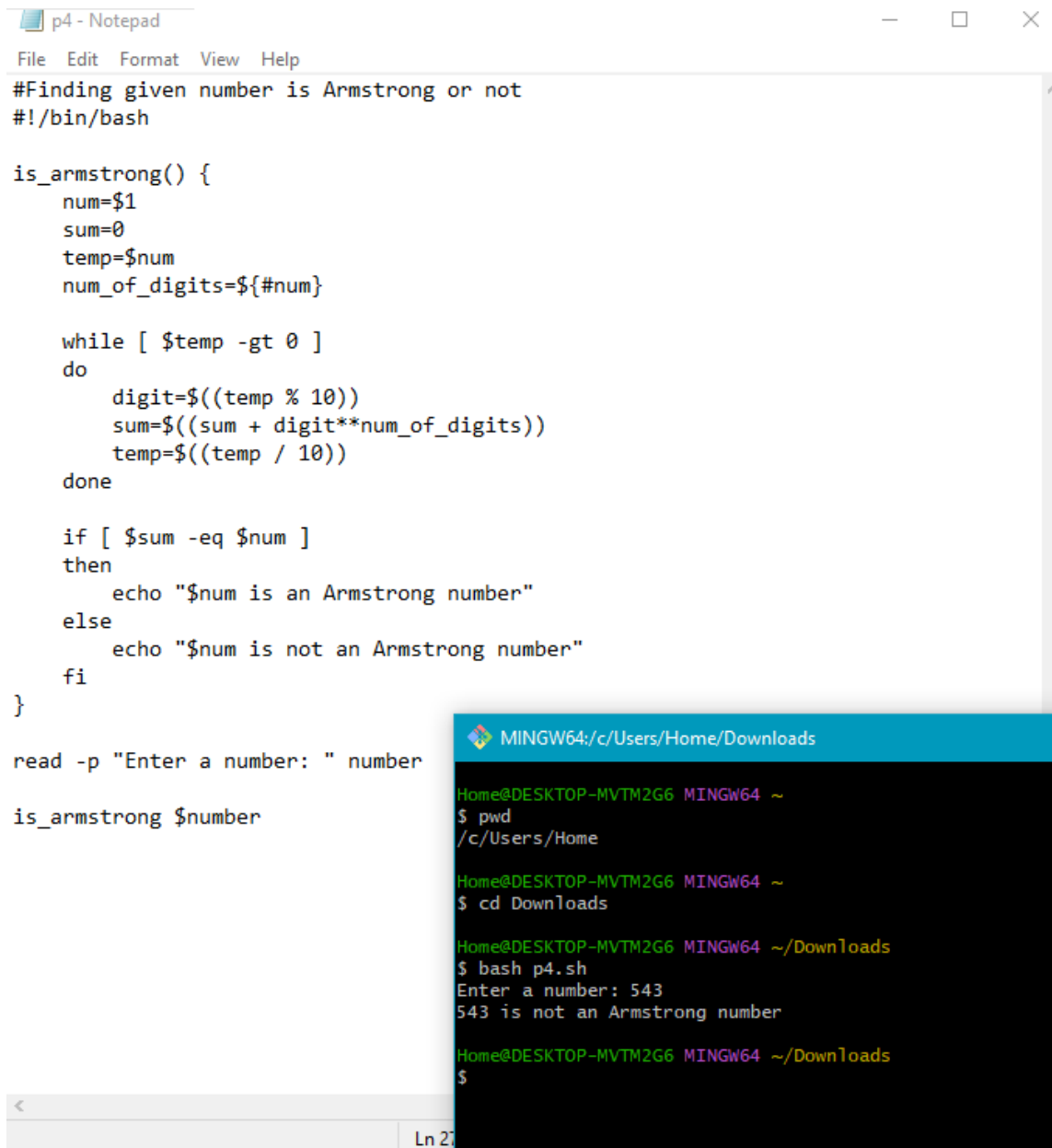
echo "The factorial of $number is: $factorial"
```



```
MINGW64: c:/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c:/Users/Home
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash pro3.sh
Enter a number: 8
The factorial of 8 is: 40320
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ |
```

PROGRAM 15:

CODE :



The image shows a Notepad window titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help). The text inside is a shell script for checking if a number is an Armstrong number. Below the Notepad window, a terminal window titled 'MINGW64:/c/Users/Home/Downloads' is open, showing the execution of the script. The terminal output shows the current directory as /c/Users/Home, then ~/Downloads, and the script being run as 'bash p4.sh'. It prompts for a number, and the user enters '543'. The script outputs '543 is not an Armstrong number'.

```
p4 - Notepad
File Edit Format View Help
#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
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

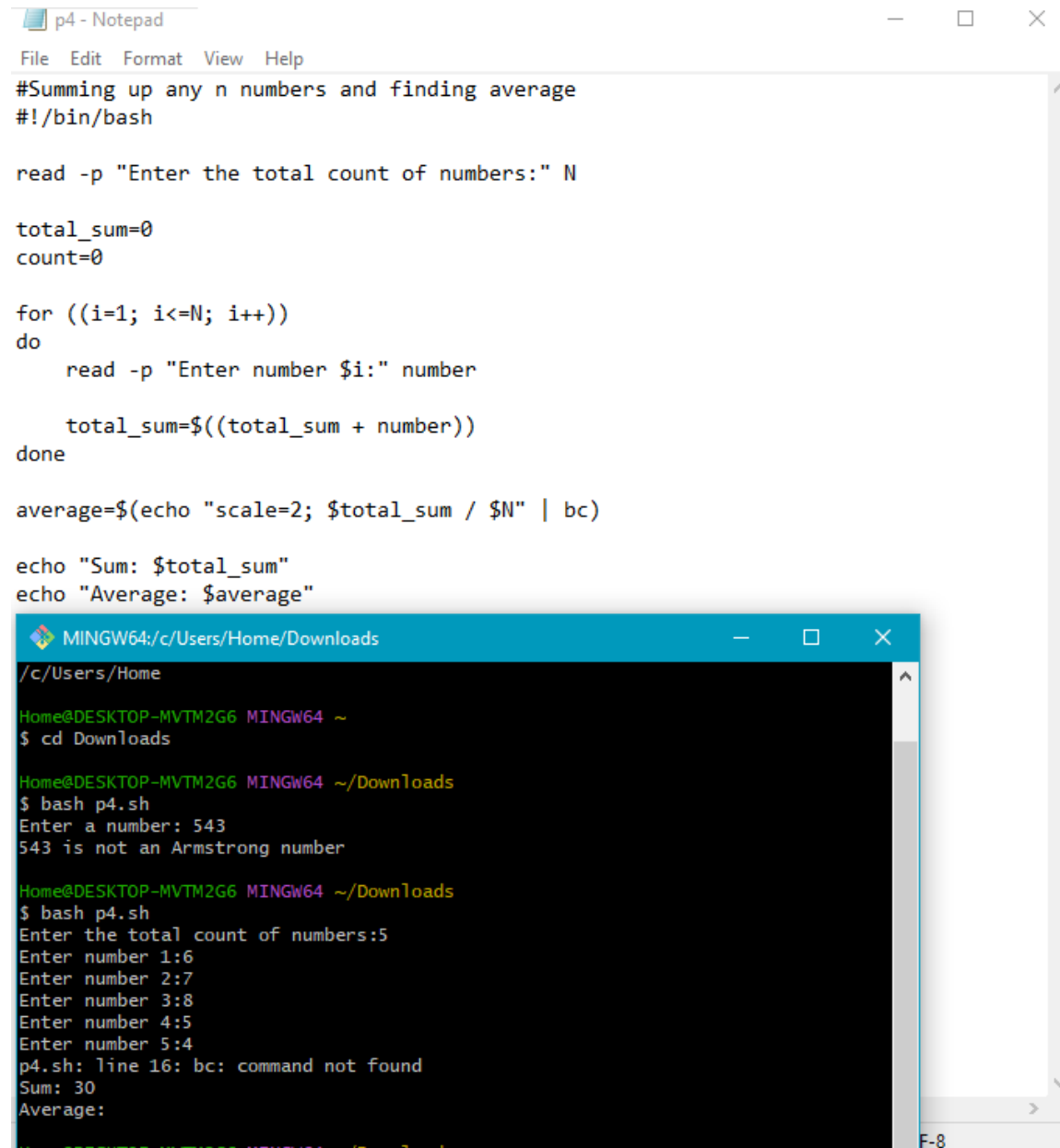
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter a number: 543
543 is not an Armstrong number

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 16 :

CODE :



The image shows a Notepad window titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help). The script content is as follows:

```
#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"
```

Below the Notepad window is a terminal window titled 'MINGW64:/c/Users/Home/Downloads'. It shows the execution of the script:

```
/c/Users/Home
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

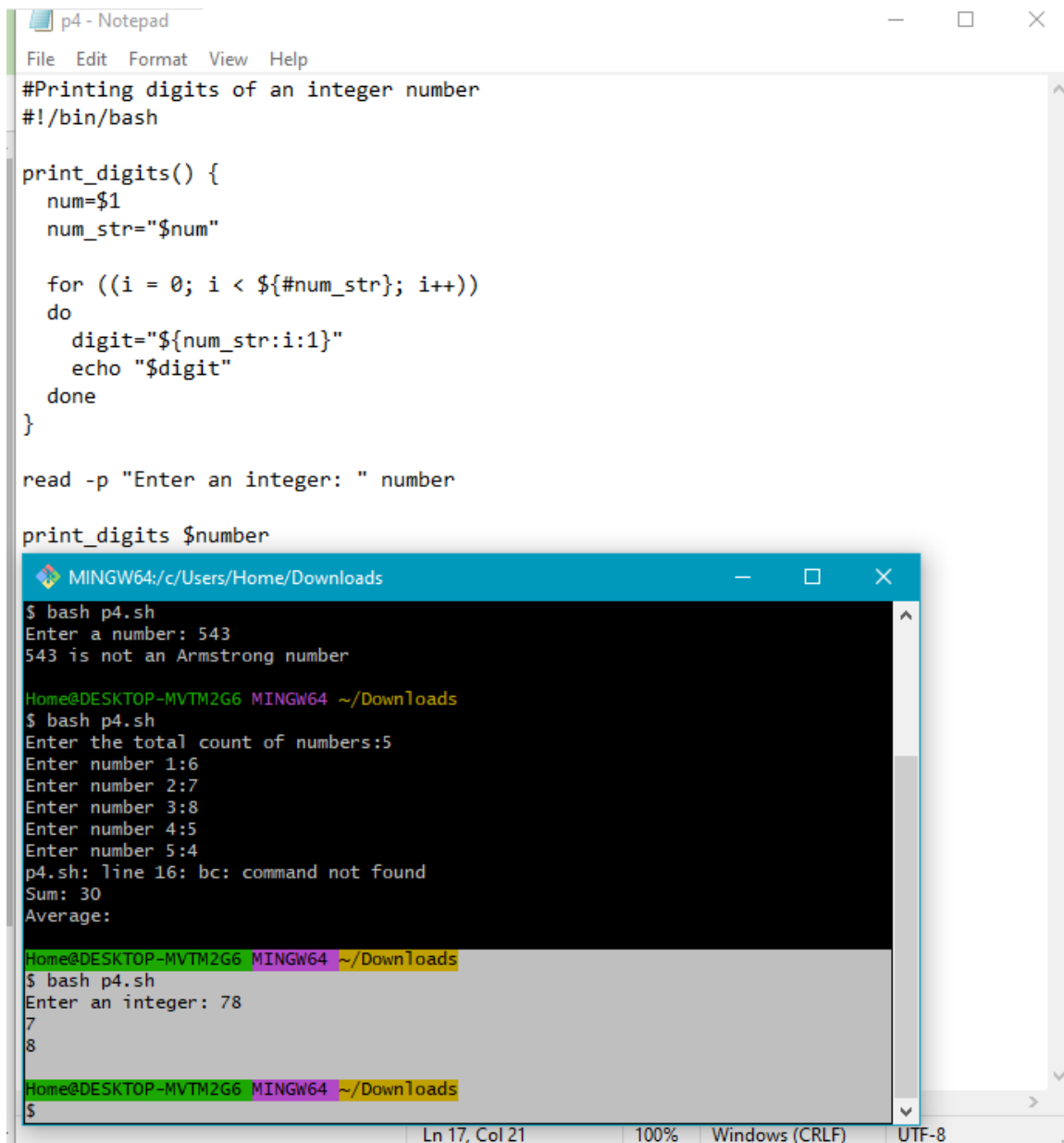
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter a number: 543
543 is not an Armstrong number

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the total count of numbers:5
Enter number 1:6
Enter number 2:7
Enter number 3:8
Enter number 4:5
Enter number 5:4
p4.sh: line 16: bc: command not found
Sum: 30
Average:
```

The terminal window has a status bar at the bottom right showing 'F-8'.

PROGRAM 17:

CODE :



The image shows a Notepad window titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help). The text inside is a shell script for printing digits of an integer. Below the script, a terminal window titled 'MINGW64/c/Users/Home/Downloads' is overlaid, showing the script's execution. The terminal shows three runs: first with input 543, then a loop of five numbers (1:6, 2:7, 3:8, 4:5, 5:4) with a 'bc: command not found' error, and finally with input 78.

```
p4 - Notepad
File Edit Format View Help
#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
```

MINGW64/c/Users/Home/Downloads

```
$ bash p4.sh
Enter a number: 543
543 is not an Armstrong number

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the total count of numbers:5
Enter number 1:6
Enter number 2:7
Enter number 3:8
Enter number 4:5
Enter number 5:4
p4.sh: line 16: bc: command not found
Sum: 30
Average:

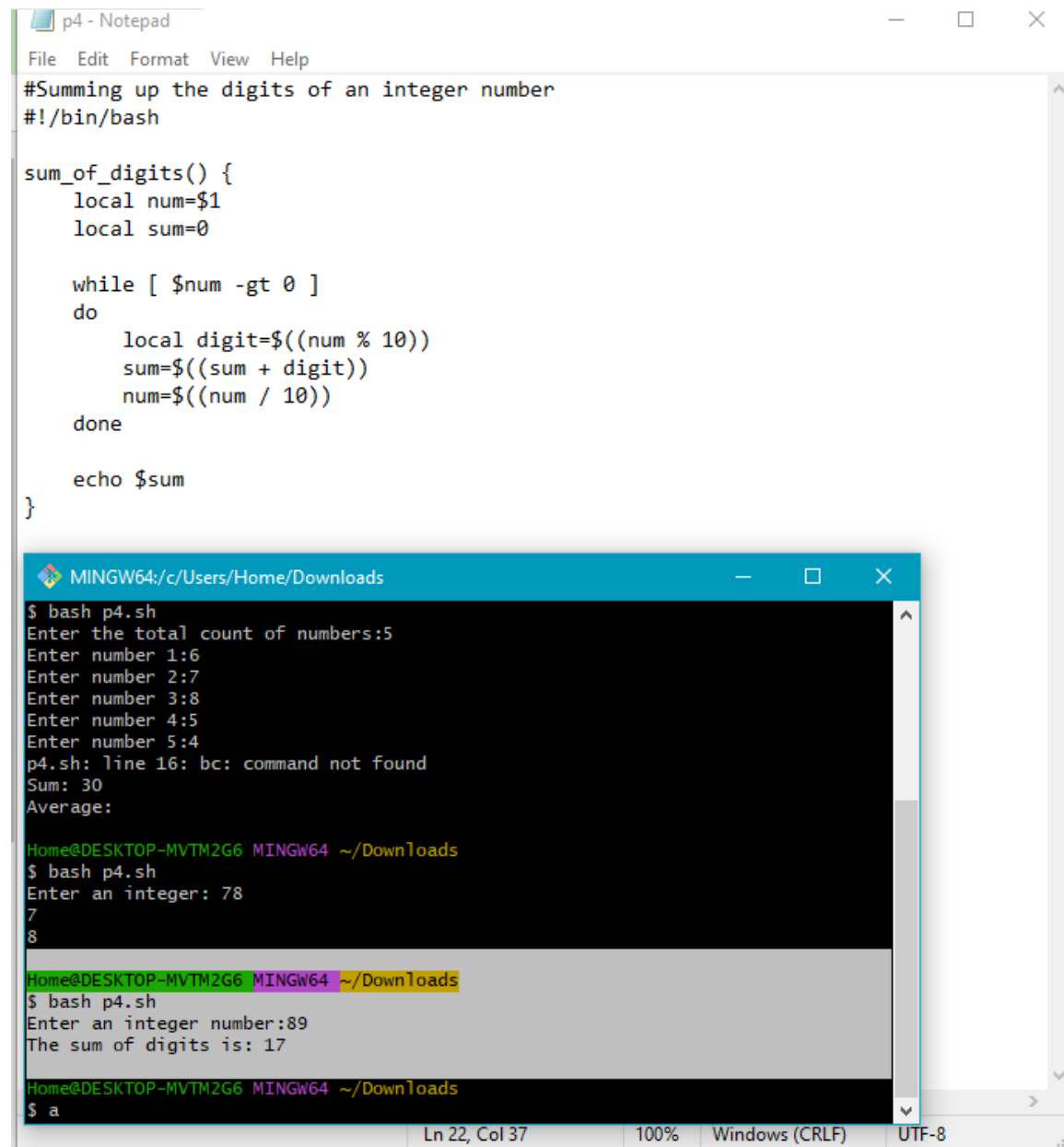
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer: 78
7
8

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 17, Col 21 | 100% | Windows (CRLF) | UTF-8

PROGRAM 18:

CODE:



The image shows two windows. The top window is a Notepad application titled 'p4 - Notepad'. It contains a shell script for summing the digits of an integer. The script uses a function 'sum_of_digits()' which takes a number as input and calculates the sum of its digits by repeatedly dividing by 10 and taking the remainder. The bottom window is a MINGW64 terminal. It shows the execution of the script 'p4.sh'. The first run prompts for a total count of numbers (5) and five numbers (6, 7, 8, 5, 4), resulting in a sum of 30. The second run prompts for an integer (78) and outputs the sum of its digits (17). The terminal window title is 'MINGW64:/c/Users/Home/Downloads'.

```
p4 - Notepad
File Edit Format View Help

#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
}

MINGW64:/c/Users/Home/Downloads
$ bash p4.sh
Enter the total count of numbers:5
Enter number 1:6
Enter number 2:7
Enter number 3:8
Enter number 4:5
Enter number 5:4
p4.sh: line 16: bc: command not found
Sum: 30
Average:

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer: 78
7
8

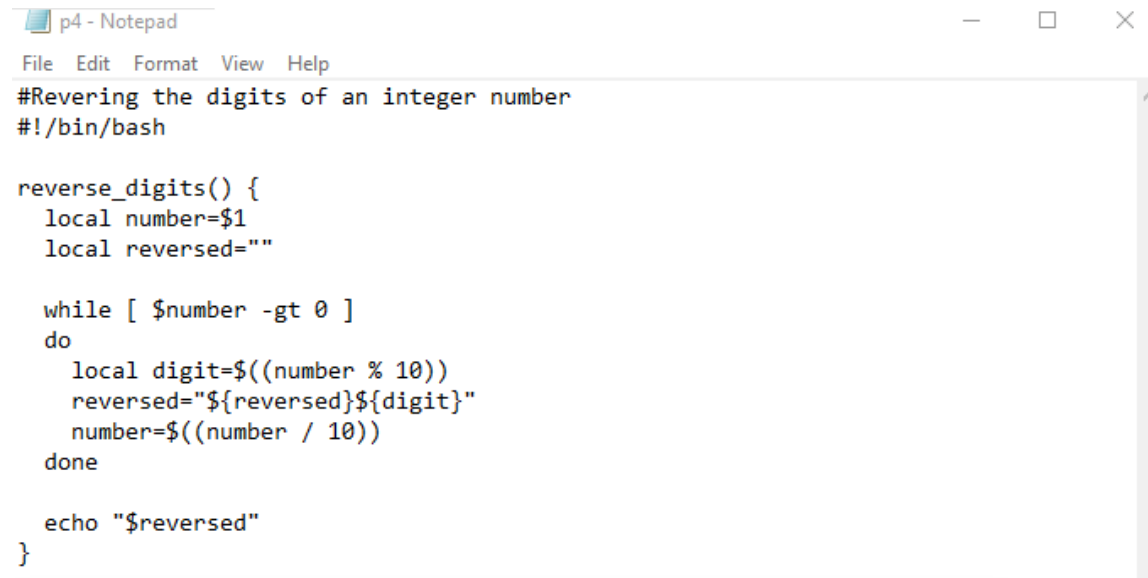
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number:89
The sum of digits is: 17

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ a
```

Ln 22, Col 37 100% Windows (CRLF) UTF-8

PROGRAM 19:

CODE :

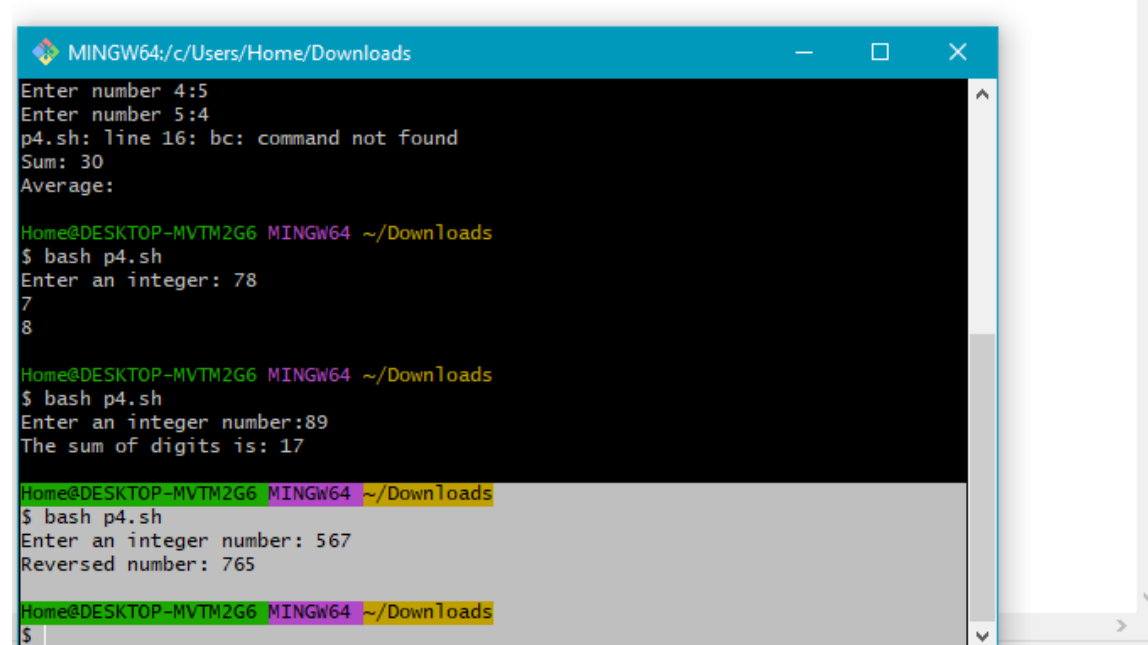


```
p4 - Notepad
File Edit Format View Help
#Revering the digits of an integer number
#!/bin/bash

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

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

    echo "$reversed"
}
```



```
MINGW64:/c/Users/Home/Downloads
Enter number 4:5
Enter number 5:4
p4.sh: line 16: bc: command not found
Sum: 30
Average:

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer: 78
7
8

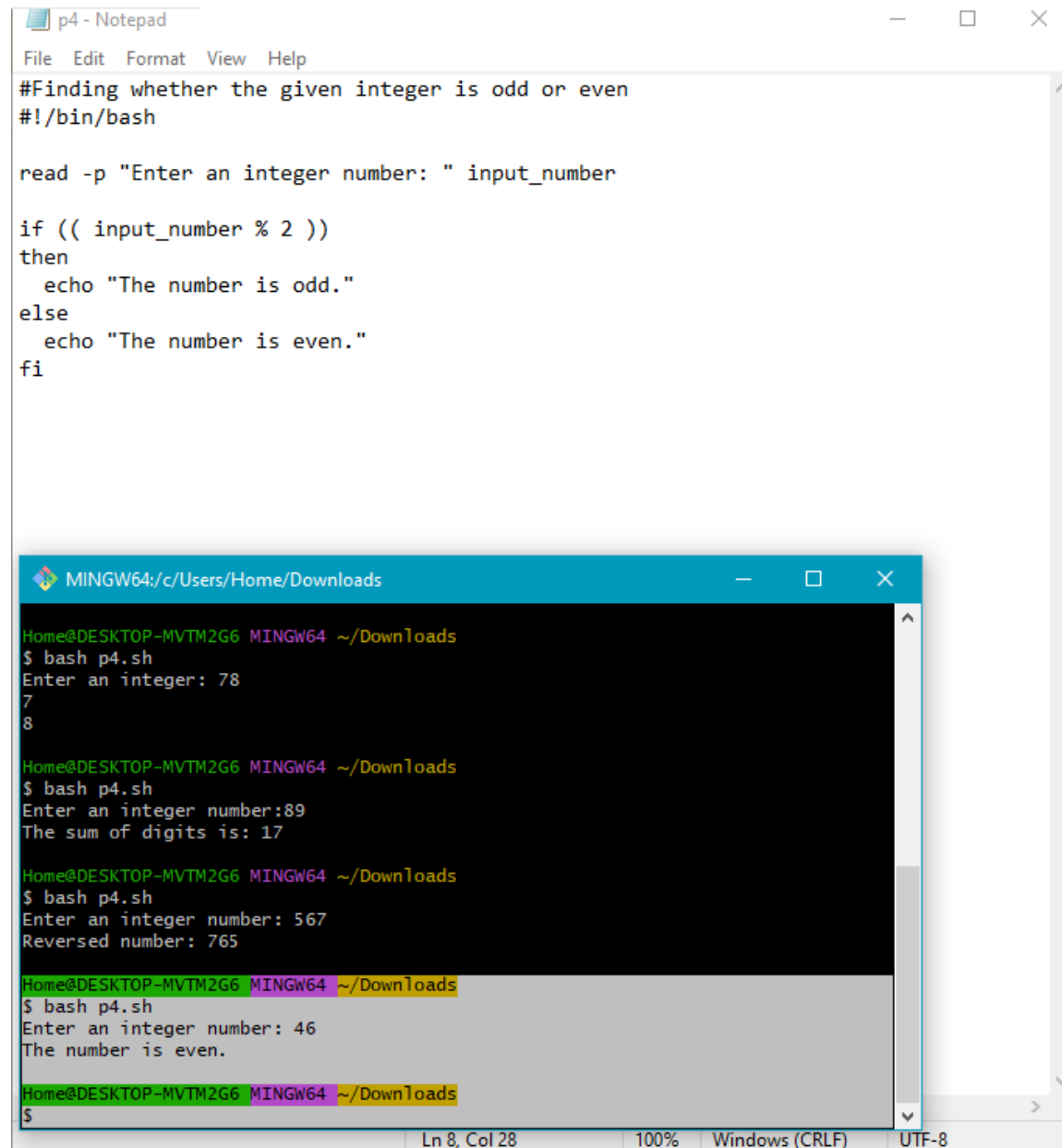
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number:89
The sum of digits is: 17

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 567
Reversed number: 765

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 20 :

CODE :



The image shows two windows. The top window is a Notepad editor titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help). It contains a shell script for checking if a number is odd or even. The script uses 'read' to get input, an 'if' statement with modulo 2 to check for odd/even, and 'echo' to print the result. The bottom window is a MINGW64 terminal titled 'MINGW64:/c/Users/Home/Downloads'. It shows the script being executed multiple times with different inputs: 78, 89, 567, and 46. The output for 89 includes a sum of digits, and for 567, it shows the reversed number. The status bar at the bottom indicates 'Ln 8, Col 28', '100%', 'Windows (CRLF)', and 'UTF-8'.

```
p4 - Notepad
File Edit Format View Help
#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
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer: 78
7
8

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number:89
The sum of digits is: 17

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 567
Reversed number: 765

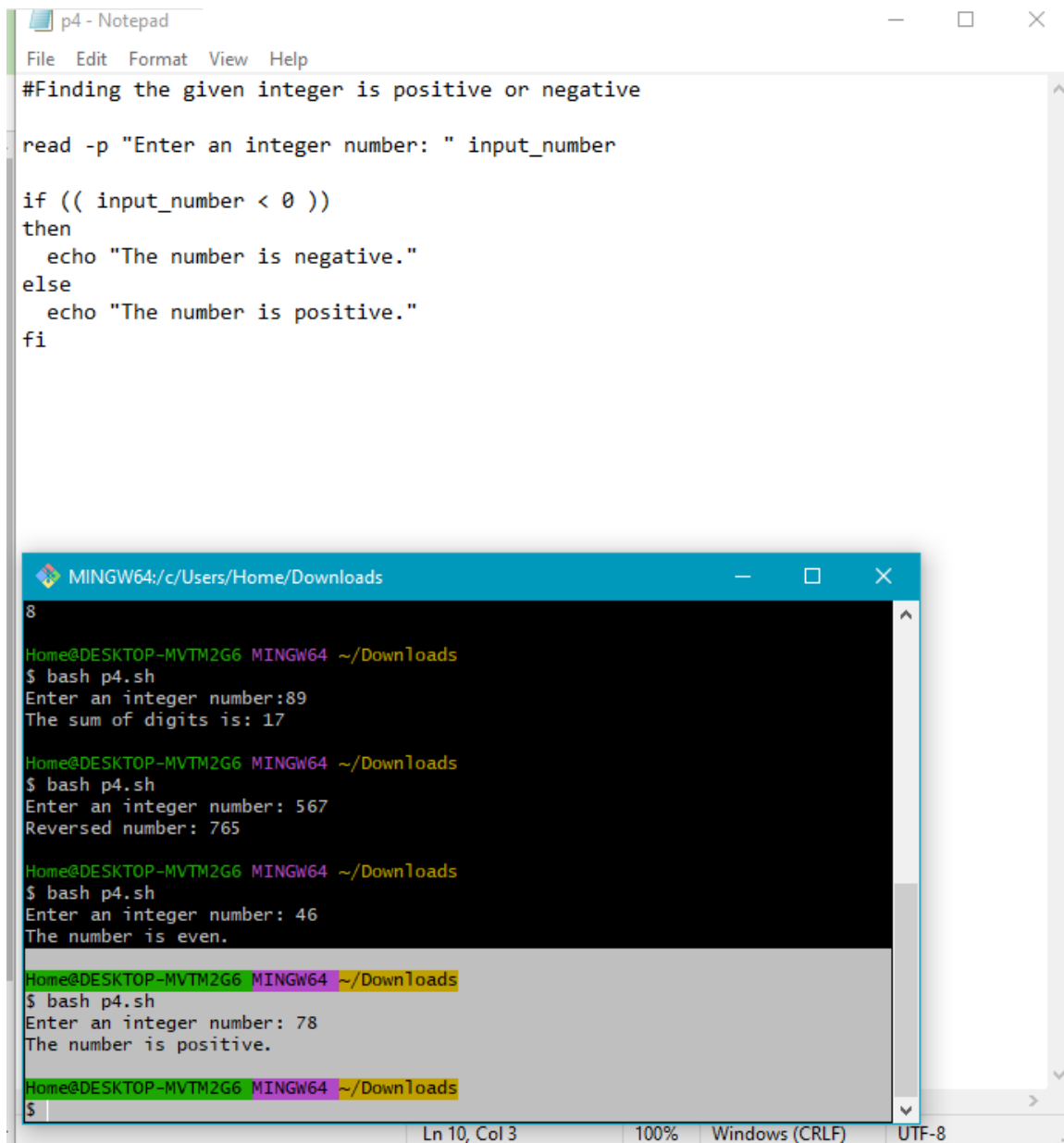
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 46
The number is even.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 8, Col 28 100% Windows (CRLF) UTF-8

PROGRAM 21 :

CODE :



The image shows two overlapping windows. The top window is a Notepad application titled 'p4 - Notepad'. It contains a shell script for checking if a number is positive or negative. The script uses 'read' to get input, an 'if' statement to check if the input is less than 0, and 'echo' to print the result. The bottom window is a terminal window titled 'MINGW64/c/Users/Home/Downloads'. It shows the execution of the script 'p4.sh' with three different inputs: 89, 567, and 46. The output for 89 is 'The sum of digits is: 17'. The output for 567 is 'Reversed number: 765'. The output for 46 is 'The number is even.'.

```
p4 - Notepad
File Edit Format View Help
#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

MINGW64/c/Users/Home/Downloads
8
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number:89
The sum of digits is: 17

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 567
Reversed number: 765

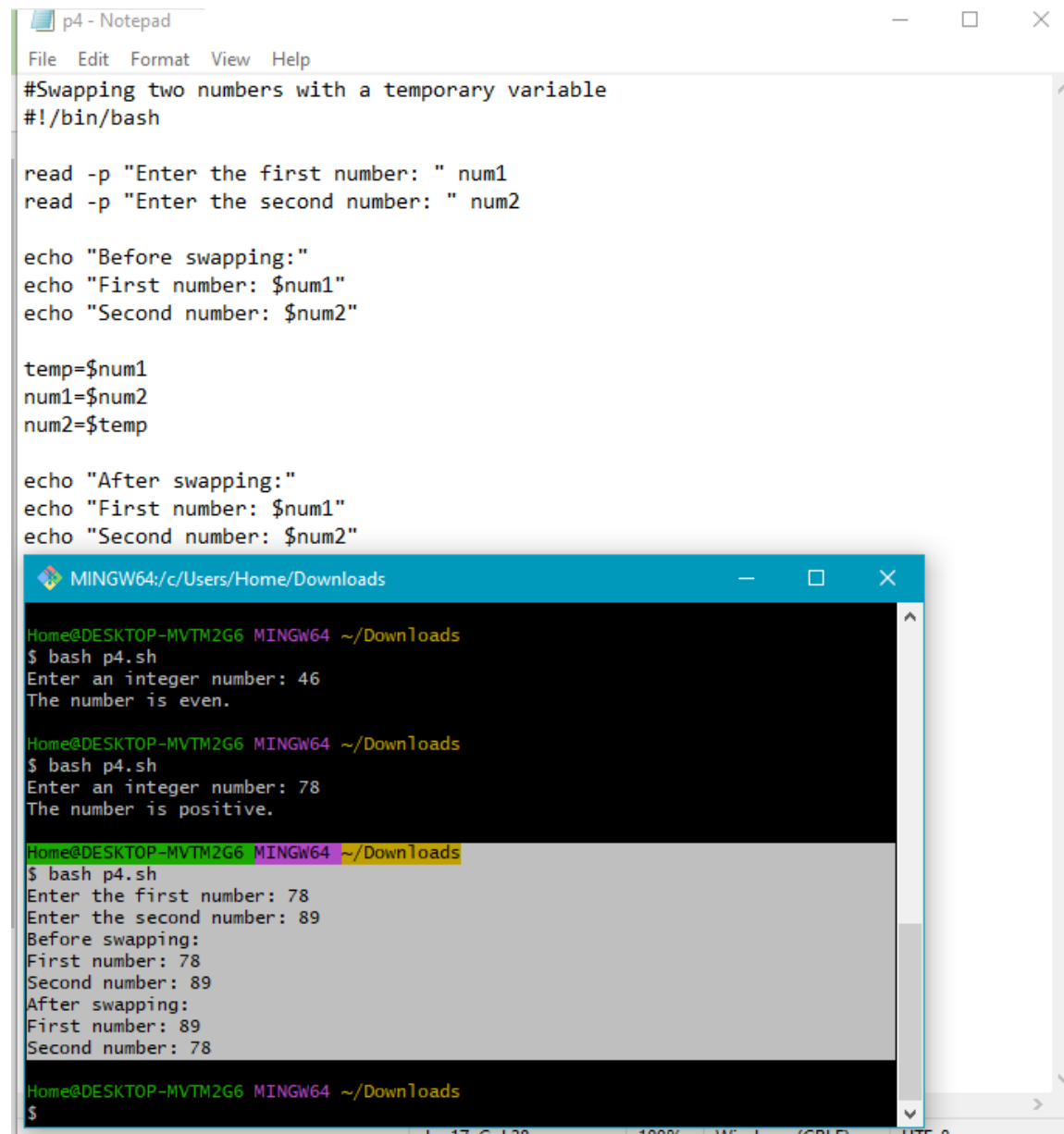
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 46
The number is even.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 78
The number is positive.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 22:

CODE:



The image shows a Notepad window titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help) and a shell script. Below it is a terminal window titled 'MINGW64: c:/Users/Home/Downloads' showing the execution of the script 'p4.sh' with three different inputs: 46, 78, and a pair of numbers (78, 89) for swapping.

```
p4 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64: c:/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 46
The number is even.

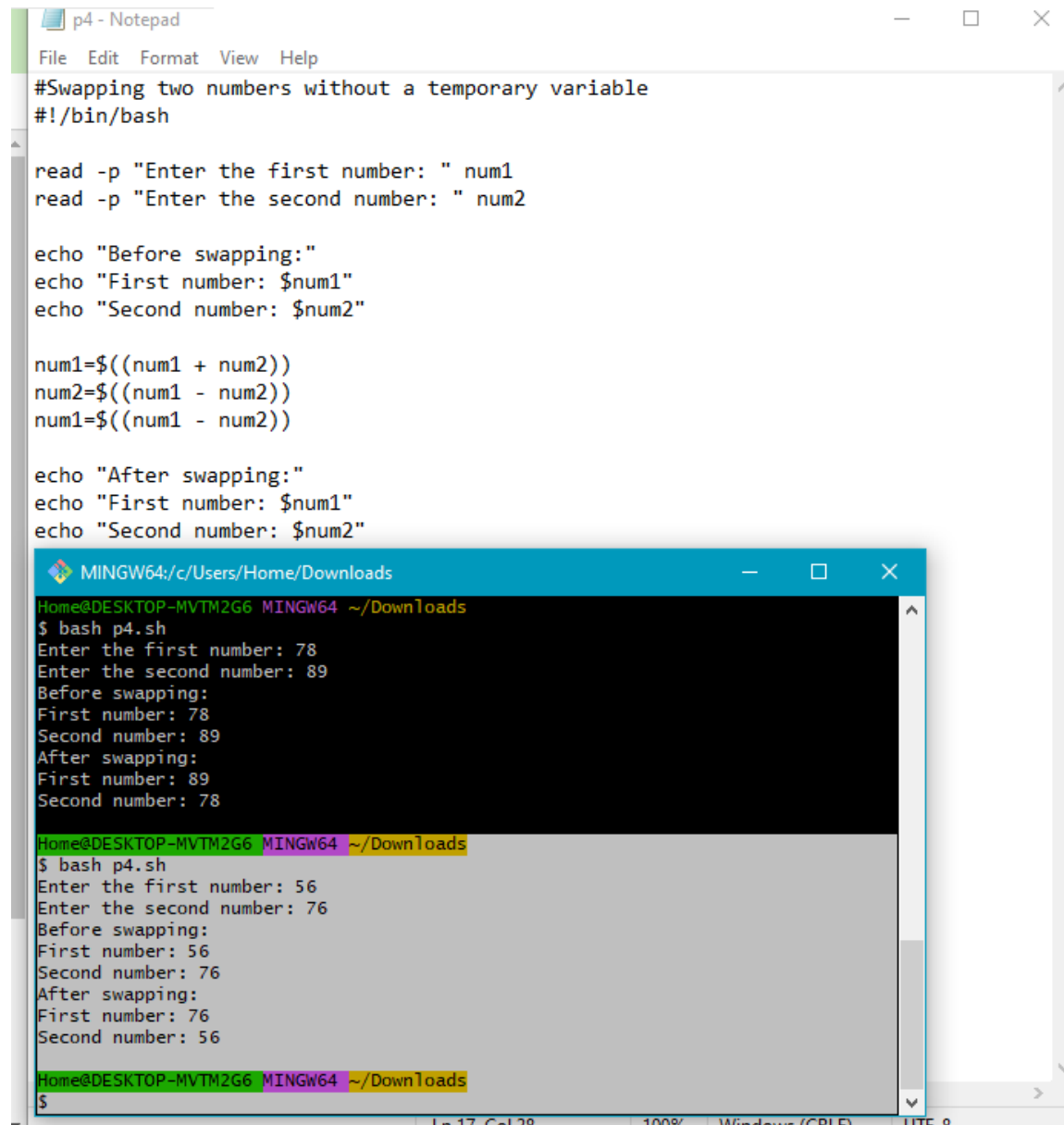
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter an integer number: 78
The number is positive.

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the first number: 78
Enter the second number: 89
Before swapping:
First number: 78
Second number: 89
After swapping:
First number: 89
Second number: 78

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 23:

CODE :



The image shows a Notepad window titled 'p4 - Notepad' containing a shell script. Below it, a terminal window titled 'MINGW64/c/Users/Home/Downloads' shows the script being executed twice. The script prompts for two numbers, displays them before and after swapping, and uses arithmetic operations to perform the swap without a temporary variable.

```
p4 - Notepad
File Edit Format View Help
#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"
```

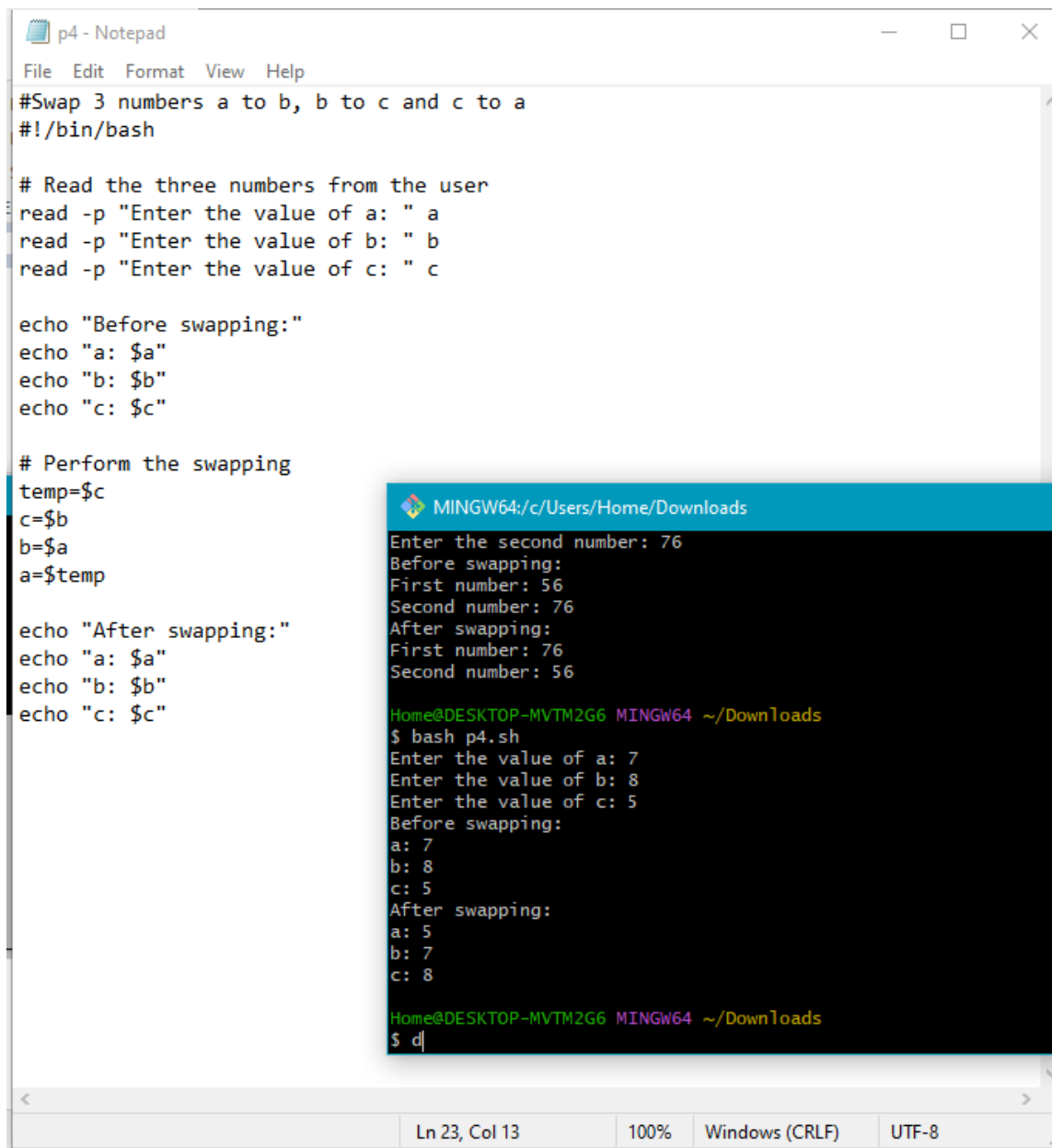
```
MINGW64/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the first number: 78
Enter the second number: 89
Before swapping:
First number: 78
Second number: 89
After swapping:
First number: 89
Second number: 78

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the first number: 56
Enter the second number: 76
Before swapping:
First number: 56
Second number: 76
After swapping:
First number: 76
Second number: 56

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

PROGRAM 24 :

CODE :



The image shows a Notepad window titled 'p4 - Notepad' containing a shell script. The script is designed to swap the values of three variables: a, b, and c. It uses a temporary variable 'temp' to store the value of 'c' before swapping it with 'b', and then 'a' with 'b'. The script prompts the user to enter values for a, b, and c, and then displays the values before and after the swapping process.

```
#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"
```

Below the Notepad window, a terminal window titled 'MINGW64:/c/Users/Home/Downloads' shows the execution of the script. The user enters the values 7, 8, and 5 for a, b, and c respectively. The terminal output shows the values before and after the swapping process, confirming that the values have been correctly swapped.

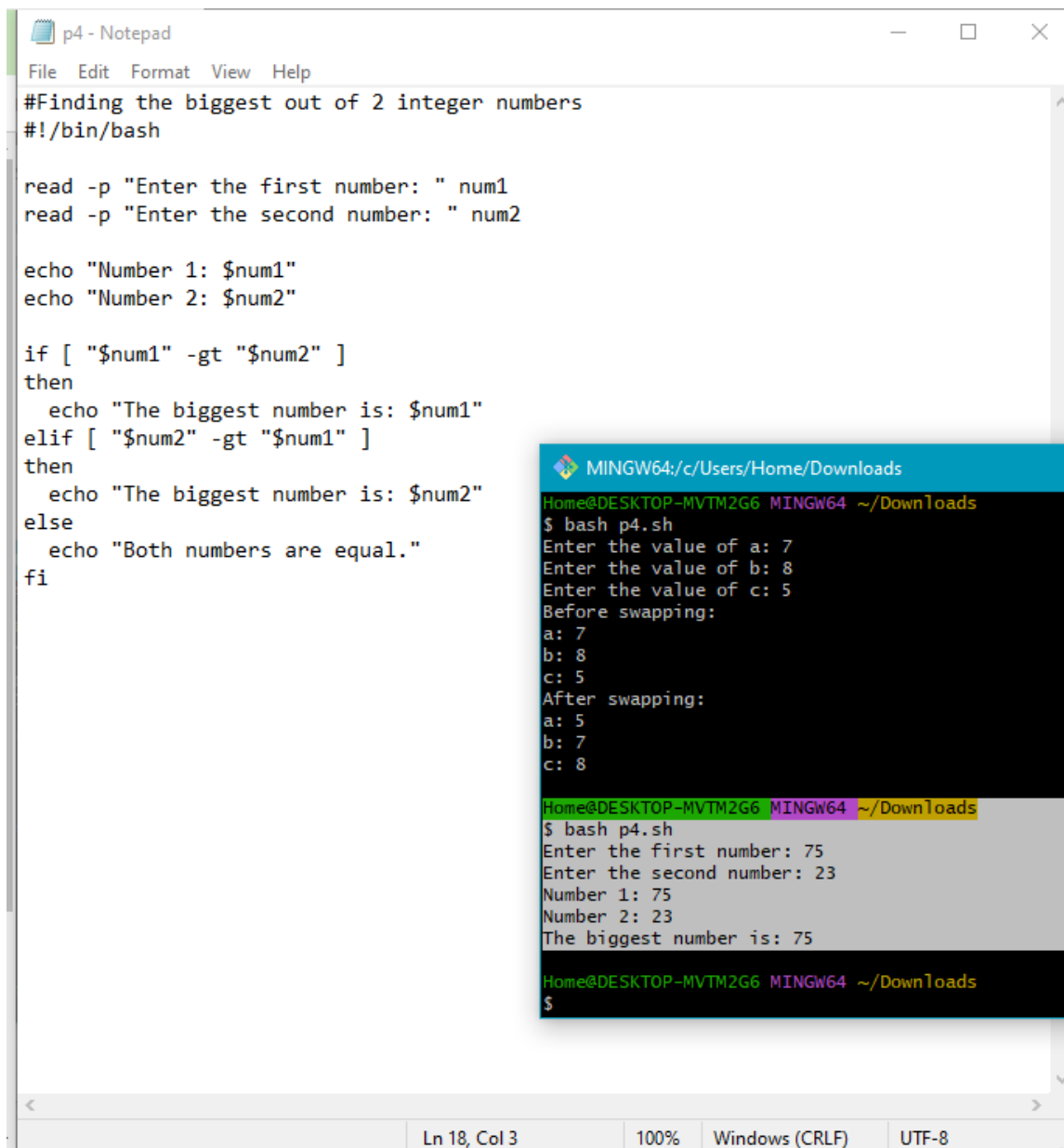
```
Enter the second number: 76
Before swapping:
First number: 56
Second number: 76
After swapping:
First number: 76
Second number: 56

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the value of a: 7
Enter the value of b: 8
Enter the value of c: 5
Before swapping:
a: 7
b: 8
c: 5
After swapping:
a: 5
b: 7
c: 8

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ d|
```

PROGRAM 25:

CODE :



The image shows a Notepad window titled 'p4 - Notepad' with a menu bar (File, Edit, Format, View, Help) and a text area containing a shell script. The script is designed to find the maximum of two integers. It uses 'read' to take input, 'echo' to display the input, and an 'if' statement with '-gt' to compare the two numbers. A second terminal window is overlaid on the right, showing the script being executed twice. The first execution takes inputs 7 and 8, and the second takes 75 and 23. The status bar at the bottom of the Notepad window indicates 'Ln 18, Col 3', '100%', 'Windows (CRLF)', and 'UTF-8'.

```
p4 - Notepad
File Edit Format View Help
#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"

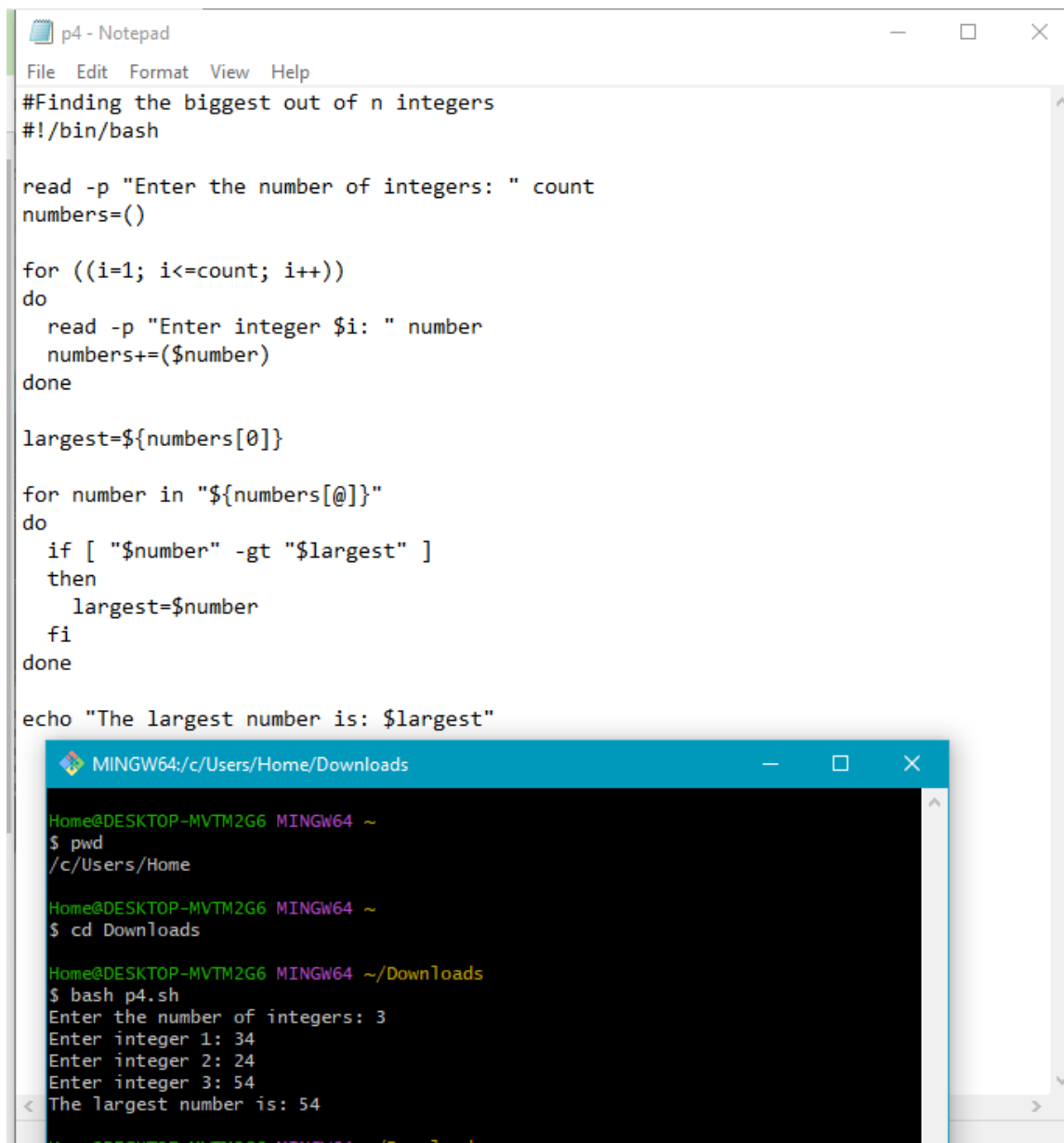
if [ "$num1" -gt "$num2" ]
then
    echo "The biggest number is: $num1"
elif [ "$num2" -gt "$num1" ]
then
    echo "The biggest number is: $num2"
else
    echo "Both numbers are equal."
fi
```

MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
\$ bash p4.sh
Enter the value of a: 7
Enter the value of b: 8
Enter the value of c: 5
Before swapping:
a: 7
b: 8
c: 5
After swapping:
a: 5
b: 7
c: 8
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
\$ bash p4.sh
Enter the first number: 75
Enter the second number: 23
Number 1: 75
Number 2: 23
The biggest number is: 75
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
\$

Ln 18, Col 3 | 100% | Windows (CRLF) | UTF-8

PROGRAM 26 :

CODE:



The image shows two overlapping windows. The top window is a Notepad application titled 'p4 - Notepad'. It contains a shell script designed to find the largest number among a set of integers. The script uses a loop to read integers from the user and compares them to find the maximum. The bottom window is a terminal application titled 'MINGW64:/c/Users/Home/Downloads'. It shows the execution of the script 'p4.sh'. The user enters 3 as the number of integers, followed by three integers: 34, 24, and 54. The script outputs 'The largest number is: 54'.

```
p4 - Notepad
File Edit Format View Help
#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"
```

```
MINGW64:/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home

Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads

Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the number of integers: 3
Enter integer 1: 34
Enter integer 2: 24
Enter integer 3: 54
The largest number is: 54
```

PROGRAM 30:

CODE:

```
p4 - Notepad
File Edit Format View Help
#Linear Search
#!/bin/bash

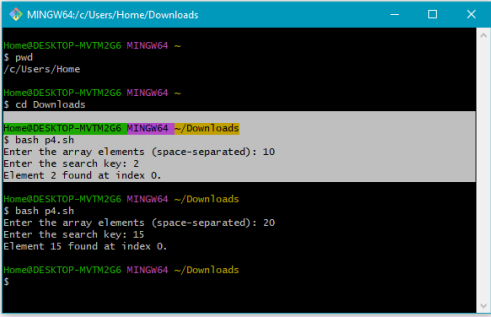
# Function to perform linear search
linear_search() {
    local search_key=$1
    # Convert function arguments into an array
    local array=("$@")
    local found=false
    local index=0

    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[@]}"
```



```
MINGW64/c/Users/Home/Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ pwd
/c/Users/Home
Home@DESKTOP-MVTM2G6 MINGW64 ~
$ cd Downloads
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the array elements (space-separated): 10
Enter the search key: 2
Element 2 found at index 0.
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$ bash p4.sh
Enter the array elements (space-separated): 20
Enter the search key: 15
Element 15 found at index 0.
Home@DESKTOP-MVTM2G6 MINGW64 ~/Downloads
$
```

Ln 33, Col 48 100% Windows (CRLF) UTF-8
32°C Mostly sunny ENG 10:22 AM

