DAY 7 ASSIGNMENT

1. Write a program in the following steps

      a. Generates 10 Random 3 Digit number.

      b. Store this random numbers into a array.

      c. Then find the 2nd largest and the 2nd smallest element without sorting the array.

PROGRAM:

```bash
#!/bin/bash
max=0
min=999

for (( i=0; i<10; i ))
do
x=0
y=0
a=$(( RANDOM%1000 ))
if [ $a -gt 99 ];
then
if [ $a -gt $max ] && [ $x -eq 0 ];
then
max2=$max
max=$a
x=1
fi
if [ $a -le $min ] && [ $y -eq 0 ];
then
min2=$min
min=$a
y=1
fi

arr[${#arr[@]}]="$a"

if [ $a -gt $max ];
then
max2=$max
max=$a
fi
if [ $a -lt $min ];
then
min2=$min
min=$a
fi
(( i++ ))
fi
done

echo "The array is:"
echo ${arr[@]}
echo "The 2nd minimum is $min2"
echo "The 2nd maximum is $max2"
```

OUTPUT:

```
chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_1.sh
The array is:
521 928 177 300 751 651 276 983 167 263
The 2nd minimum is 177
The 2nd maximum is 928
```

2. Extend the above program to sort the array and then find the 2nd largest
and the 2nd smallest element.

PROGRAM:

```
MINGW64:/c/Users/chitr/BRIDGELABZ                                    —    □    ×

 GNU nano 4.9.3                          7_2.sh
#!/bin/bash

for (( i=0; i<10; i ))

do

b=$(( RANDOM%1000 ))

if [ $b -gt 99 ];
then
a[${#a[@]}]="$b"
(( i++ ))
fi
done #1
echo ""
echo "the unsorted array is:"
echo ${a[@]}

for (( c=0; c<10; c++ ))
do
for (( j=$c; j<10; j++ ))
do

if [ ${a[$c]} -gt ${a[$j]} ];
then
temp=${a[$j]}
a[$j]=${a[$c]}
a[$c]=$temp
fi

done
done
echo ""
echo "Array after sorting: "
echo ${a[@]}

echo ""
echo "The 2nd minimum  is ${a[1]} and 2nd maximum is ${a[8]}"
```

OUTPUT

```
chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_2.sh

the unsorted array is:
311 815 701 812 864 973 932 944 787 403

Array after sorting:
311 403 701 787 812 815 864 932 944 973

The 2nd minimum  is 403 and 2nd maximum is 944
```

3. Extend the Prime Factorization Program to store all the Prime Factors of a number n into an array and finally display the output.

PROGRAM:

```bash
GNU nano 4.9.3                        7_3_new1.sh
#!/bin/bash

echo "enter a number"
read no
echo ""
echo "The array of the prime factors of $n is:"
for (( p=2; p<=no; p++ ))
do
if [ $(( no%p )) -eq 0 ];
then

for (( i=2; i<p; i++ ))
do
x=$(( p%i ))
if [ $x -eq 0 ];
then
break
else
continue
fi
done

if [ $i -eq $p ];
then
arr[${arr[@]}]="$i"
fi

fi
done

echo ${arr[@]}
```

OUTPUT:

```
chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_3_new1.sh
enter a number
158

The array of the prime factors of  is:
2 79
```

4. Write a Program to show Sum of three Integer adds to ZERO

PROGRAM:

```
MINGW64:/c/Users/chitr/BRIDGELABZ                                    —    □    ×

  GNU nano 4.9.3                          7_4.sh
#!/bin/bash
sum=0
for (( i=0; i<3; i++ ))
do
        echo "enter number"
        read n
        arr[${#arr[@]}]="$n"
        sum=$(( sum+n ))
done

if [ $sum -eq 0 ];
then
echo "sum is zero"

else
echo "Sum is not zero"
fi
```
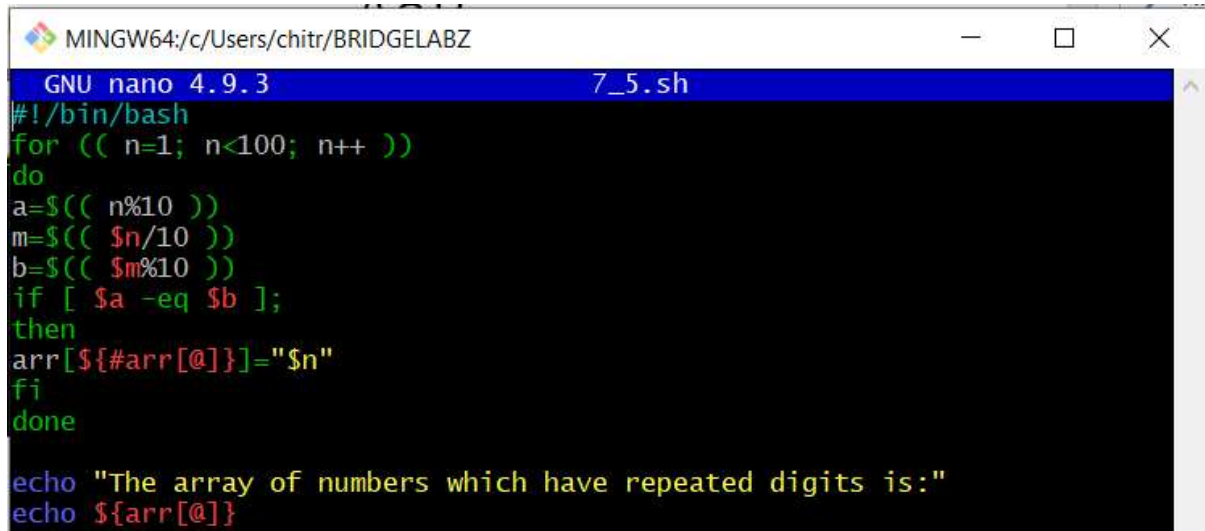
OUTPUT:

```
chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_4.sh
enter number
15
enter number
-30
enter number
15
sum is zero

chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_4.sh
enter number
15
enter number
48
enter number
79
Sum is not zero
```

5. Take a range from 0 – 100, find the digits that are repeated twice like 33, 77, etc. and store them in an array
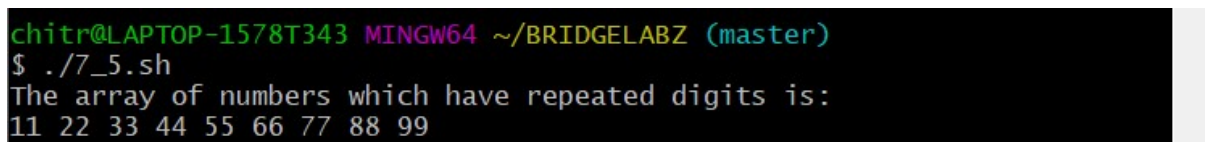
PROGRAM:

```
MINGW64:/c/Users/chitr/BRIDGELABZ                                    —    □    ×

  GNU nano 4.9.3                        7_5.sh
#!/bin/bash
for (( n=1; n<100; n++ ))
do
a=$(( n%10 ))
m=$(( $n/10 ))
b=$(( $m%10 ))
if [ $a -eq $b ];
then
arr[${#arr[@]}]="$n"
fi
done

echo "The array of numbers which have repeated digits is:"
echo ${arr[@]}
```

OUTPUT:

```
chitr@LAPTOP-1578T343 MINGW64 ~/BRIDGELABZ (master)
$ ./7_5.sh
The array of numbers which have repeated digits is:
11 22 33 44 55 66 77 88 99
```