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.

```
#!/bin/bash -x
counter=0
max=0
maxSec=0
min=1000
minSec=1000
for (( i=1; i<=10; i++ ))
      do
            num=$((RANDOM%1000))
            if [ $num -lt 100 ]
                  then
                        num=$(($num+100))
                        value[((counter++))]="$num"
            else
                  num=$num
                  value[((counter++))]="$num"
      fi
done
for (( i=0; i<10; i++ ))
      do
            x=${value[$i]}
            if [ $x -gt $max ]
                  then
                              max=$x;
            elif [[ $x -lt $max && $x -gt $maxSec ]]
                  then
                        maxSec=$x
                  else
                        max=$max
                        maxSec=$maxSec
            fi
            echo $max
            echo $maxSec
      done
for (( i=0; i<10; i++ ))
      do
            x=${value[$i]}
            if [ $x -lt $min ]
                  then
```

```
                              min=$x;
            elif [[ $x -gt $min && $x -lt $minSec ]]
                  then
                        minSec=$x
                  else
                        min=$min
                        minSec=$minSec
            fi
            echo $min
            echo $minSec
      done
echo ${value[@]}
echo $max
echo $maxSec
echo $min
echo $minSec
```

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

```
#!/bin/bash -x
counter=0
max=0
maxSec=0
min=1000
minSec=1000
for (( i=1; i<=10; i++ ))
      do
            num=$((RANDOM%1000))
            if [ $num -lt 100 ]
                  then
                        num=$(($num+100))
                        value[((counter++))]="$num"
                  else
                        num=$num
                        value[((counter++))]="$num"
      fi
done
for ((i = 0; i<10; i++))
do

    for((j = 0; j<10-i-1; j++))
    do

        if [ ${value[j]} -gt ${value[$((j+1))]} ]
        then
            # swap
            temp=${value[$j]}
            value[$j]=${value[$((j+1))]}
            value[$((j+1))]=$temp
        fi
    done
done
echo "Array in sorted order :"
echo ${value[*]}
echo maxSec=${value[8]}
echo minSec=${value[1]}
```

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

```
#!/bin/bash -x
counter=0
read -p "enter number for which factor " x;
for (( i=2; $x>1; i++ ))
    do
        z=$(($x%$i))
        while [ $z -eq 0 ]
            do
            factors[((counter++))]=$i

            x=$(($x/$i))
            z=$(($x%$i))
            done
    done
echo ${factors[@]}
```

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

```
#!/bin/bash
read -p "enter the integers count(min 3) " numberOfInt
count=0
for (( i=1; i<=$numberOfInt; i++ ))
    do
        read -p "enter value for integer " int[$((i-1))]
    done
z=$(($numberOfInt-2))
for (( j=0; j<$z; j++))
    do
```

```
        y=$(($numberOfInt-1))
        for (( k=$(($j+1)); k<$y; k++ ))
            do
                for(( l=$k+1; l<$numberOfInt; l++ ))
                    do
                        if [ $(( int[$j] + int[$k] + int[$l] )) -eq 0 ]
                            then
                                ((count++))
                                echo ${int[$j]} ${int[$k]} ${int[$l]}
                        fi
                    done
            done
    done
if [ $count -eq 0 ]
then
    echo "no Such Triplet"
fi
```

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

```
#!/bin/bash -x
counter=0
for (( i=10; i<=100; i++ ))
      do
            n=$i
            r=0
            while [[ $n -ne 0 ]]
                  do
                        r=$r*10
                        r=$(($r+$n%10))
                        n=$(($n/10))
                  done
            echo $r
                        if [ $r -eq $i ]
                              then
                                    double[((counter++))]="$r"
                        fi
      done
echo ${double[@]}
```