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

declare -a randoms

for ((count=0;count<10;count++))

do

randoms[$count]=$((RANDOM%900+100))

done

echo "ARRAY GENERATED: " ${randoms[@]}

largest=0

largest2=0

smallest=999

smallest2=999

for num in ${randoms[@]}

do

if [ $num -gt $largest ]

then

largest=$num

fi

done

for num2 in ${randoms[@]}

do

if [ $num2 -gt $largest2 -a $num2 -ne $largest ]

then

largest2=$num2

fi

done

echo "2nd Largest Number: " $largest2

for item in ${randoms[@]}

do

if [ $item -lt $smallest ]

then

smallest=$item

fi

done

for item2 in ${randoms[@]}

do

if [ $item2 -lt $smallest2 -a $item2 -ne $smallest ]

then

smallest2=$item2

fi

done

echo "2nd Smallest Number: " $smallest2

ARRAY GENERATED: 147 557 970 398 959 353 781 307 489 448

2nd Largest Number: 959

2nd Smallest Number: 307

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

and the 2nd smallest element.

#! /bin/bash

declare -a randoms

for ((count=0;count<10;count++))

do

randoms[$count]=$((RANDOM%900+100))

done

echo "ARRAY GENERATED: " ${randoms[@]}

n=${#randoms[@]}

temp=0

for ((i=0;i<$n;i++))

do

for ((j=1;j<$((n-i));j++))

do

if [ ${randoms[$((j-1))]} -gt ${randoms[$j]} ]

then

temp=${randoms[$((j-1))]}

randoms[$((j-1))]=${randoms[$j]}

randoms[$j]=$temp

fi

done

done

echo "SORTED ARRAY: "${randoms[@]}

echo "2nd Largest Number: " ${randoms[8]}

echo "2nd Smallest Number: " ${randoms[1]}

ARRAY GENERATED: 590 776 312 701 995 436 625 170 344 483

SORTED ARRAY: 170 312 344 436 483 590 625 701 776 995

2nd Largest Number: 776

2nd Smallest Number: 312

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

function primefactor(){

num=$1

flag=1

for ((i=2;i<=num;i++))

do

if [ $((num%i)) -eq 0 ]

then

for((j=2;j<i;j++))

do

if [ $((i%j)) -eq 0 ]

then

flag=0

break;

else

flag=1

fi

done

if [ $flag -eq 1 ]

then

echo $i

fi

fi

done

}

read -p "Enter a number: " num

result=$(primefactor $num)

count=0

declare -a factors

for item in $result

do

factors[$count]=$item

count=$((count+1))

done

echo "Prime Factors of" $num "is" ${factors[@]}

Enter a number: 100

Prime Factors of 100 is 2 5

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

#! /bin/bash

declare -a numbers

numbers=(3 -3 0 6 -5 -1)

n=${#numbers[@]}

for ((i=0;i<n-2;i++))

do

for ((j=i+1;j<n-1;j++))

do

for ((k=j+1;k<n;k++))

do

if [ $(( ${numbers[$i]}+${numbers[$j]}+${numbers[$k]} )) -eq 0 ]

then

echo "Sum of" ${numbers[$i]}"," ${numbers[$j]}"," ${numbers[$k]} "is ZERO"

fi

done

done

done

Sum of 3, -3, 0 is ZERO

Sum of 6, -5, -1 is ZERO

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

declare -a numbers

count=11

for ((i=0;i<100;i++))

do

if [ $i -eq $count ]

then

numbers[$i]=$i

count=$((count+11))

fi

done

echo ${numbers[@]}

11 22 33 44 55 66 77 88 99