**Day-7 Assignments**

**Arrays Practice Problems**

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.

# generate 3 digit random number

function generateRandomNumber() {

randomNumber=$((RANDOM%1000))

echo $randomNumber

}

# Find the second largest number

function findSecondLargest() {

randomNumbers=("$@")

largestNumber=${randomNumbers[0]}

secondLargestNumber=${randomNumbers[0]}

smallestNumber=${randomNumbers[0]}

secondSmallestNumber=${randomNumbers[0]}

length="${#randomNumbers[@]}"

for ((counter=1; counter < $length ; counter++))

do

if [ $largestNumber -lt ${randomNumbers[$counter]} ]

then

secondLargestNumber=$largestNumber

largestNumber=${randomNumbers[$counter]}

elif [ $secondLargestNumber -lt ${randomNumbers[$counter]} ]

then

secondLargestNumber=${randomNumbers[$counter]}

fi

if [ $smallestNumber -gt ${randomNumbers[$counter]} ]

then

secondSmallestNumber=$smallestNumber

smallestNumber=${randomNumbers[$counter]}

elif [ $secondSmallestNumber -gt ${randomNumbers[$counter]} ]

then

secondSmallestNumber=${randomNumbers[$counter]}

fi

done

echo "Second Largest number is $secondLargestNumber"

echo "Second Smallest number is $secondSmallestNumber"

}

# Store the random number to the unsorted array

function storeRandomNumber() {

counter=0

while [ $counter -lt 10 ]

do

randomNumberArray[((counter++))]=$(generateRandomNumber)

done

echo ${randomNumberArray[@]}

findSecondLargest ${randomNumberArray[@]}

}

storeRandomNumber

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

# generate 3 digit random number

function generateRandomNumber() {

randomNumber=$((RANDOM%1000))

echo $randomNumber

}

# Find the second largest number

function sortTheArray() {

randomNumbers=("$@")

length="${#randomNumbers[@]}"

for (( counterOne = 0; counterOne < ${#randomNumbers[@]}; counterOne++ ))

do

for (( counterTwo = $counterOne; counterTwo < ${#randomNumbers[@]}; counterTwo++ ))

do

if [ ${randomNumbers[$counterOne]} -gt ${randomNumbers[$counterTwo]} ]; then

temp=${randomNumbers[$counterOne]}

randomNumbers[$counterOne]=${randomNumbers[$counterTwo]}

randomNumbers[$counterTwo]=$temp

fi

done

done

echo "Sorted array is " ${randomNumbers[@]}

echo "Second Largest number is ${randomNumbers[ $(( ${#randomNumbers[@]}-2 )) ]}"

echo "Second Smallest number is ${randomNumbers[1]}"

}

# Store the random number to the unsorted array

function storeRandomNumber() {

counter=0

while [ $counter -lt 10 ]

do

randomNumberArray[((counter++))]=$(generateRandomNumber)

done

echo ${randomNumberArray[@]}

sortTheArray ${randomNumberArray[@]}

}

storeRandomNumber

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

read -p "Enter the number to find factors:" number

declare a primeFactors

counter=0

primeFactorsOf=$number

for ((count=2 ; count<= $number ; count++ ))

do

while (($number%$count == 0 ))

do

primeFactors[((counter++))]=$count

number=$((number/count))

done

done

echo "Prime factors of $primeFactorsOf is ${primeFactors[@]}"

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

echo "Enter theree number: "

for (( i=0 ; i<3; i++ ))

do

read -p "Enter the number one by one: " num

arr[i]=$num

done

sum=0

for i in ${num[@]}

do

total=`expr $sum + $i`

done

echo $sum

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

declare -a repeatedNumberArray

countArray=0

for ((counter=10 ; counter < 100 ; counter++ ))

do

if [ $((counter%10)) -eq $(((counter/10)%10)) ]

then

repeatedNumberArray[((countArray++))]=$counter

fi

done

echo "Repeated number is ${repeatedNumberArray[@]} "