DAILY ONLINE ACTIVITIES SUMMARY

Date:	**01-06-2020**	**Name:**	**Anvitha Poojary**
Sem & Sec	**6A**	**USN:**	**4AL17CS008**

Online Test Summary

Subject	**CNSC**		
Max. Marks	**60**	**Score**	**48**

Certification Course Summary

Course	Step into Robotic Process Automation		
Certificate Provider	**GUVI**	**Duration**	**3hr**

Coding Challenges

Problem Statement:

1.Python Program to remove duplicate elements from a list

2.Write a Java Program to left rotate the elements of an array

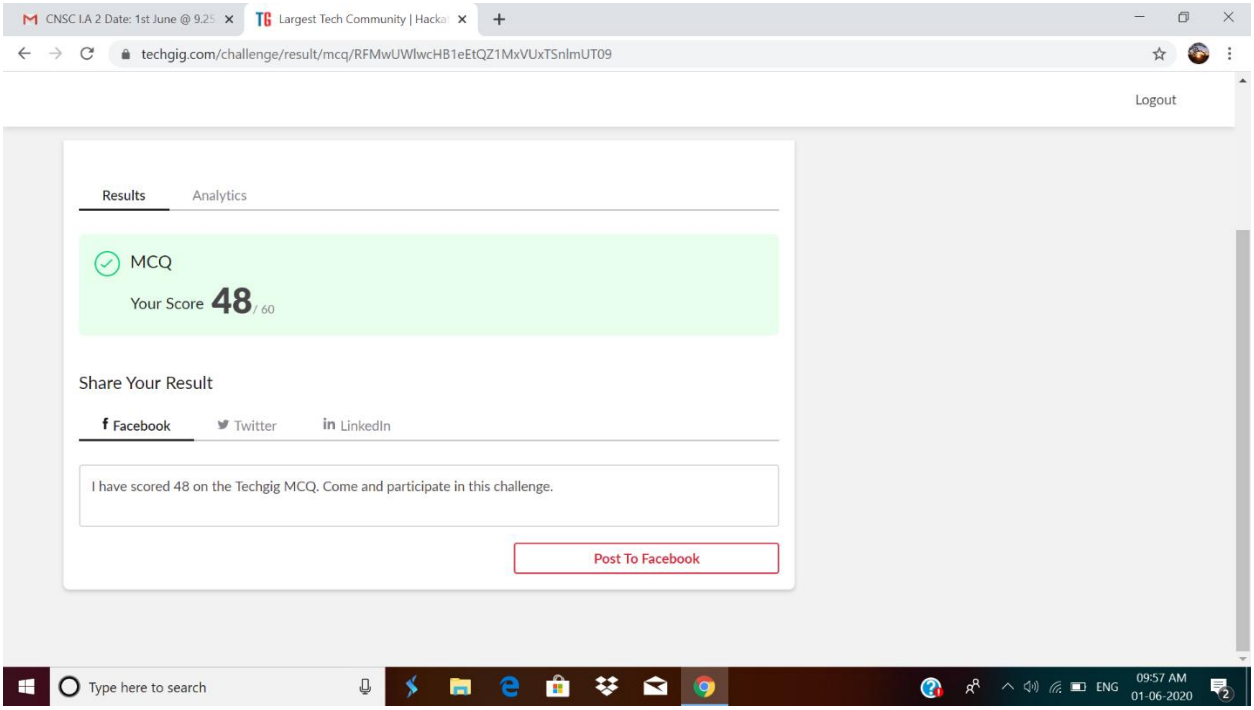
3. Given an array of positive integers. Write a C Program to find the leaders in the array.

Status: completed

Uploaded the report in Github	**Yes**
If yes Repository name	https://github.com/anvithapo99/Daily-Report
Uploaded the report in slack	**Yes**

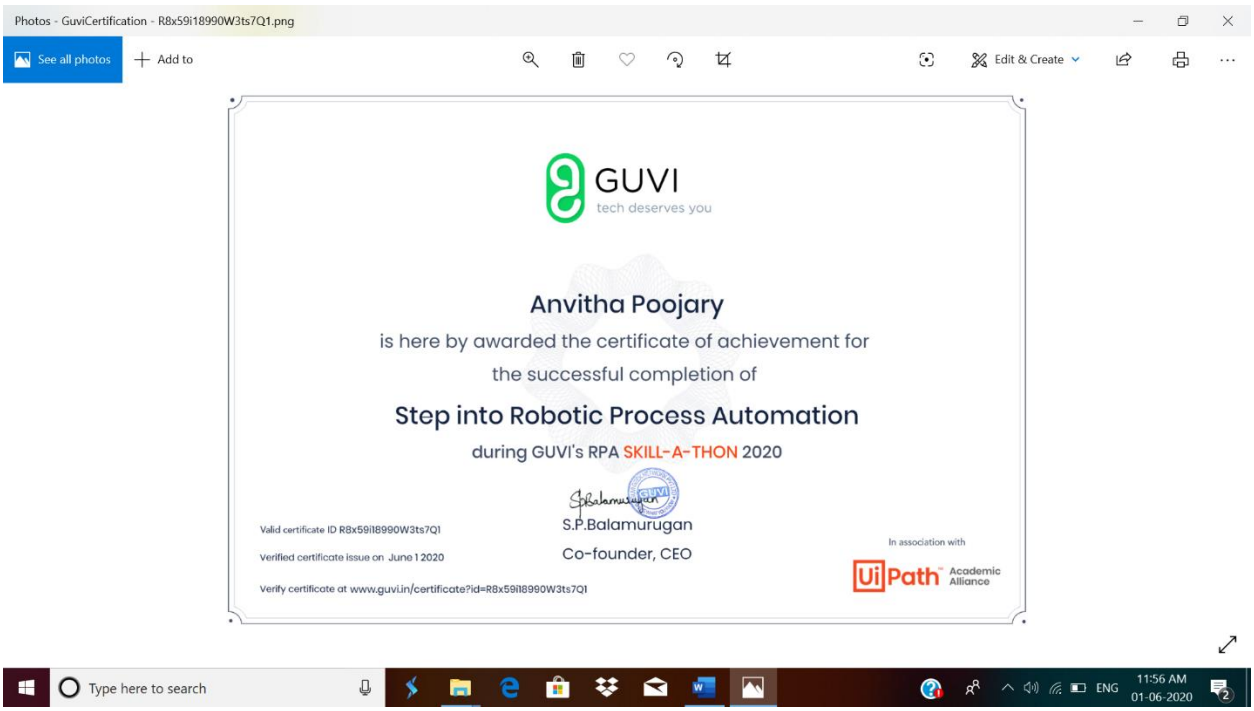
Online test details:

Subject: CNSC



Certification course details:

Step into Robotic Process Automation



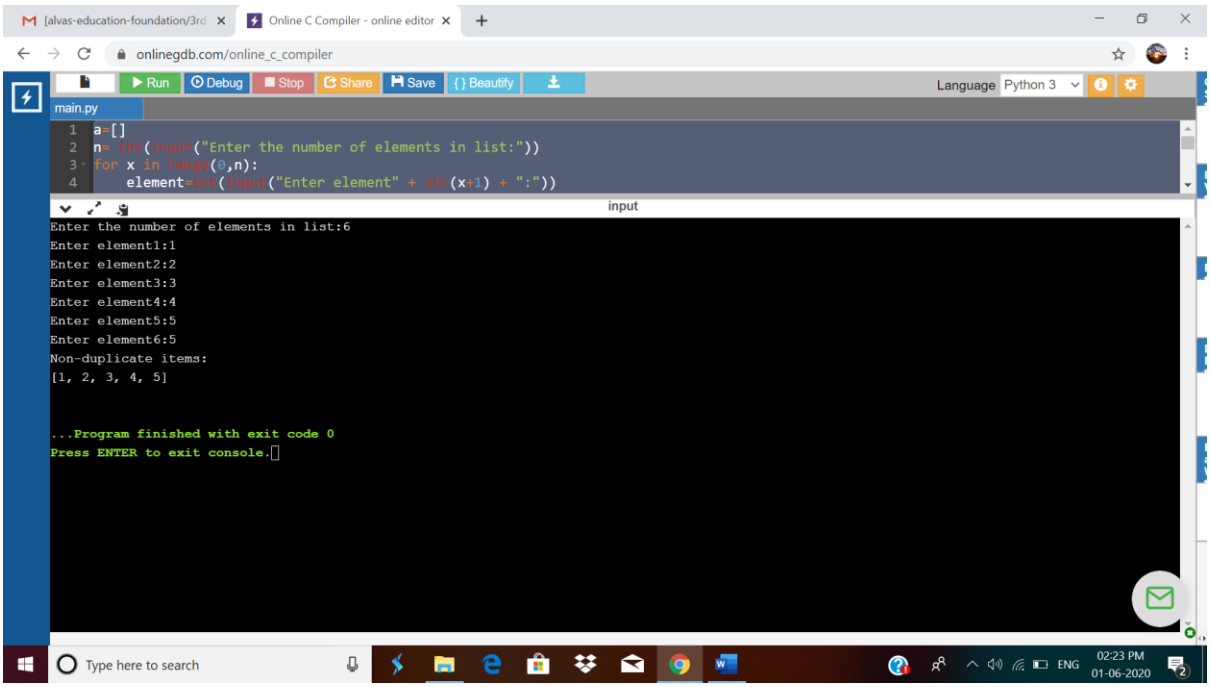
Coding Challenges Details:

1.Python Program to remove duplicate elements from a list

```
a=[]
n= int(input("Enter the number of elements in list:"))
for x in range(0,n):
    element=int(input("Enter element" + str(x+1) + ":"))
    a.append(element)
b = set()
unique = []
for x in a:
    if x not in b:
        unique.append(x)
```

```
    b.add(x)
print("Non-duplicate items:")
print(unique)
```

output:



2.Write a Java Program to left rotate the elements of an array

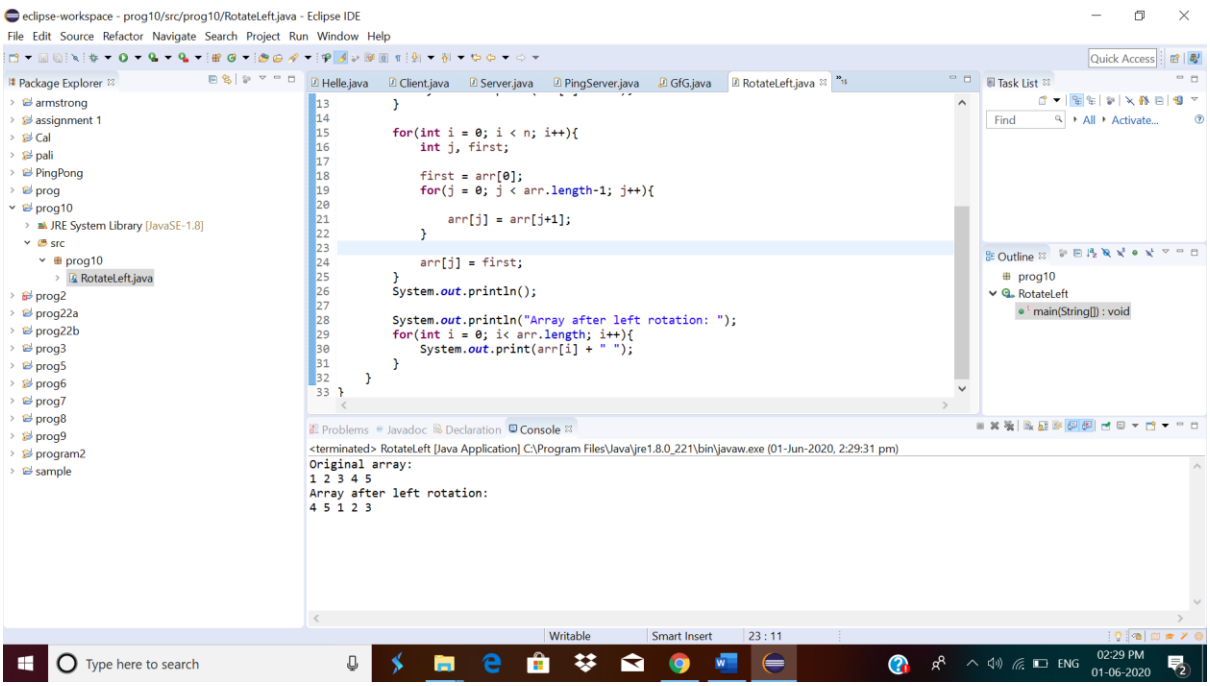
Problem Description
In this program, we need to rotate the elements of an array towards the left by the specified number of times. In the left rotation, each element of the array will be shifted to its left by one position and the first element of the array will be added to end of the list. This process will be followed for a specified number of times.
Suppose if n is 1 then, all elements of the array will be moved to its left by one position such that second element of the array will take the first position, the third element will be moved to the second position and so on. The first element of the array will be added to the last of the array.

Algorithm
STEP 1: START

STEP 2: INITIALIZE arr[] ={1, 2, 3, 4, 5 }.
STEP 3: SET n =3
STEP 4: PRINT "Original Array"
STEP 5: REPEAT STEP 6 for(i=0; i<arr.length; i++)
STEP 6: PRINT arr[i]
STEP 7: REPEAT STEP 8 to STEP 12 for(i=0; i<n; i++)
STEP 8: DEFINE j, first.
STEP 9: first = arr[0]
STEP 10: REPEAT STEP 11 for(j= 0; j<arr.length-1; j++)
STEP 11: arr[j]= arr[j+1]
STEP 12: arr[j]= first
STEP 13: PRINT "Array after left rotation"
STEP 14: REPEAT STEP 15 for(i=0; i<arr.length; i++)
STEP 15: PRINT arr[i]
STEP 16: END

```
package prog10;

class RotateLeft {
    public static void main(String[] args) {

        int [] arr = new int [] {1, 2, 3, 4, 5};

        int n = 3;

        System.out.println("Original array: ");
        for (int i = 0; i < arr.length; i++) {
            System.out.print(arr[i] + " ");
        }

        for(int i = 0; i < n; i++){
            int j, first;

            first = arr[0];
            for(j = 0; j < arr.length-1; j++){

                arr[j] = arr[j+1];
            }

            arr[j] = first;
        }
        System.out.println();

        System.out.println("Array after left rotation: ");
        for(int i = 0; i< arr.length; i++){
            System.out.print(arr[i] + " ");
        }
    }
}
```

Output:



3. Given an array of positive integers. Write a C Program to find the leaders in the array.

Note: An element of array is leader if it is greater than or equal to all the elements to its right side. Also, the rightmost element is always a leader.

Input:
The first line of input contains an integer T denoting the number of test cases. The description of T test cases follows.
The first line of each test case contains a single integer N denoting the size of array.
The second line contains N space-separated integers A1, A2, ..., AN denoting the elements of the array.

Output:
Print all the leaders.
Constraints:
1 <= T <= 100
1 <= N <= 107
0 <= Ai <= 107

Example:

Input:
3
6

16 17 4 3 5 2
5
1 2 3 4 0
5
7 4 5 7 3

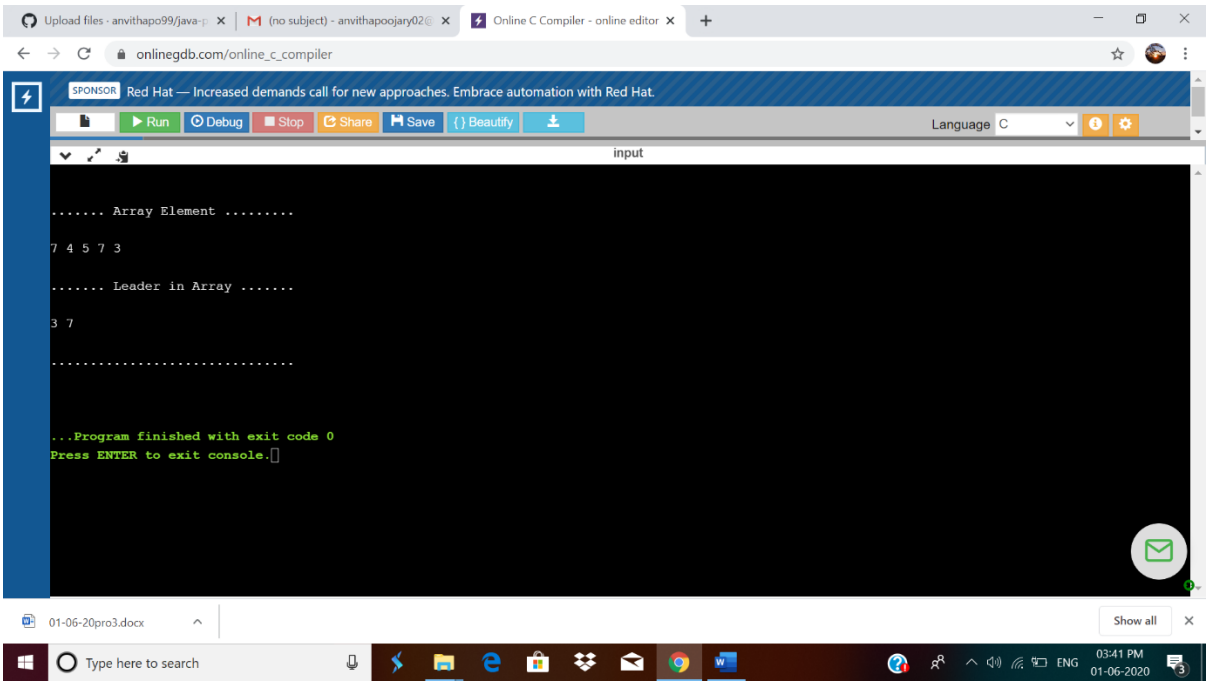
Output:
17 5 2
4 0
7 7 3

Explanation:
Testcase 3: All elements on the right of 7 (at index 0) are smaller than or equal to 7. Also, all the elements of right side of 7 (at index 3) are smaller than 7. And, the last element 3 is itself a leader since no elements are on its right.

```
#include<stdio.h>

#include<limits.h>

void ArrayLeader(int arr[],int size);

void PrintArray(int arr[],int size);

int main(void)

{

int arr[] = {7,4,5,7,3};

int size = 5;

printf("\n\n..... Array Element .....\n\n");

PrintArray(arr,size);

printf("\n\n..... Leader in Array .....\n\n");

ArrayLeader(arr,size);

printf("\n\n.....\n\n");

return 0;

}

void ArrayLeader(int arr[],int size)

{
```

```
int MaxTillNow,i;

MaxTillNow = INT_MIN;

for(i=size-1;i>=0;--i)

{

if(arr[i] > MaxTillNow)

{

printf("%d ",arr[i]);

MaxTillNow = arr[i];

}

}

}

void PrintArray(int arr[],int size)

{

int i;

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

printf("%d ",arr[i]);

}
```

Output:


Upload files - anvithapo99/java-p
(no subject) - anvithapoojary02@
Online C Compiler - online editor
onlinegdb.com/online_c_compiler
SPONSOR
Red Hat — Increased demands call for new approaches. Embrace automation with Red Hat.
Run
Debug
Stop
Share
Save
{ } Beautify
Language C
input
..... Array Element .....
7 4 5 7 3
..... Leader in Array .....
3 7
...Program finished with exit code 0
Press ENTER to exit console.
01-06-20pro3.docx
Show all
Type here to search
ENG
03:41 PM
01-06-2020