

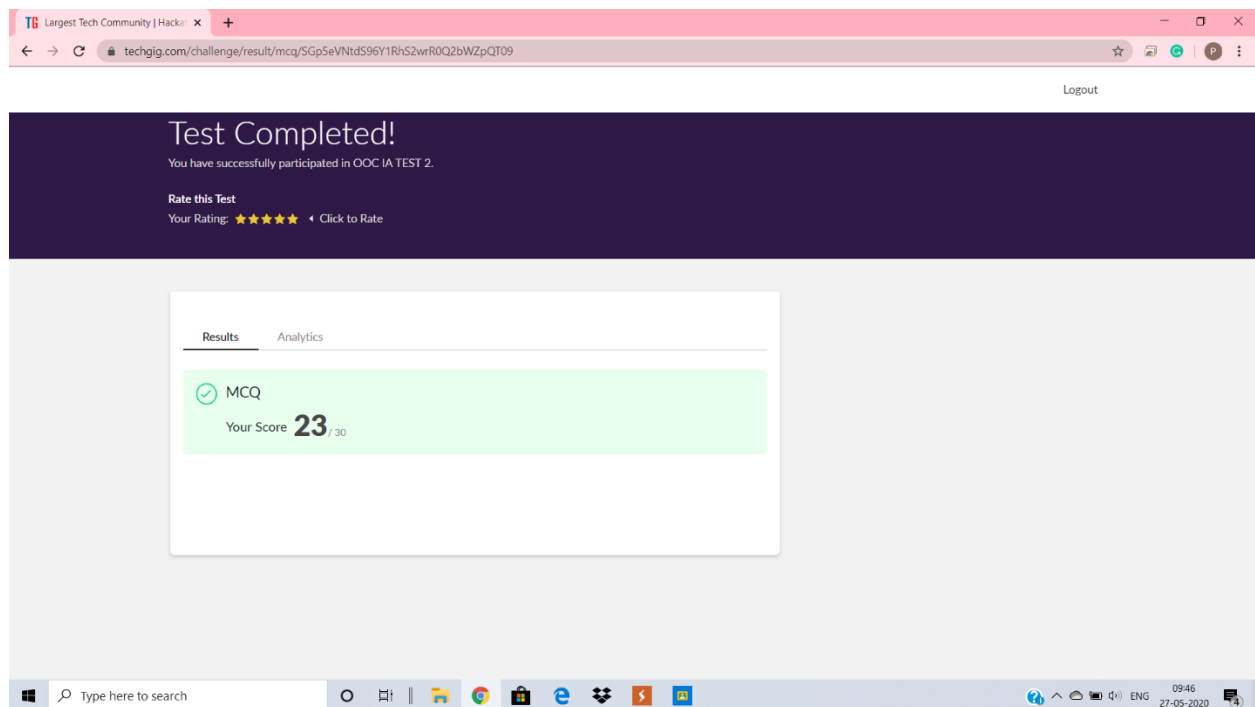
DAILY ONLINE ACTIVITIES SUMMARY

Date:	25/05/2020	Name:	Priya Nagari
Sem & Sec	Fourth SEM section B	USN:	4AL18CS063
Online Test Summary			
Subject	Object oriented concepts(18CS45)		
Max. Marks	30	Score	23
Certification Course Summary			
Course	The complete Android app development Masterclass: Build apps		
Certificate Provider	Udemy	Duration	29 hours
Coding Challenges			
Problem Statement: 1. Problem Statement: 1. Write a C Program to sort an array of integers in ascending order and Display the sorted array and Number of passes performed for sorting. 2. Given an array arr[] of the positive integers of size N, the task is to find the largest element on the left side of each index which is smaller than the element present at that index. Note: If no such element is found then print -1.			
Status:			
Uploaded the report in Github		YES	
If yes Repository name		Priya_Nagari link: https://github.com/alvas-education-foundation/Priya_Nagari	

Uploaded the report in slack	YES
------------------------------	-----

Online Test Details:

The Object oriented concepts (18CS45) 2nd Internal Assessment was conducted on 5th Module. In that I had Scored 23 marks out of 30.



Certification Course Details:

Name of the course: The complete Android app development Masterclass: Build apps

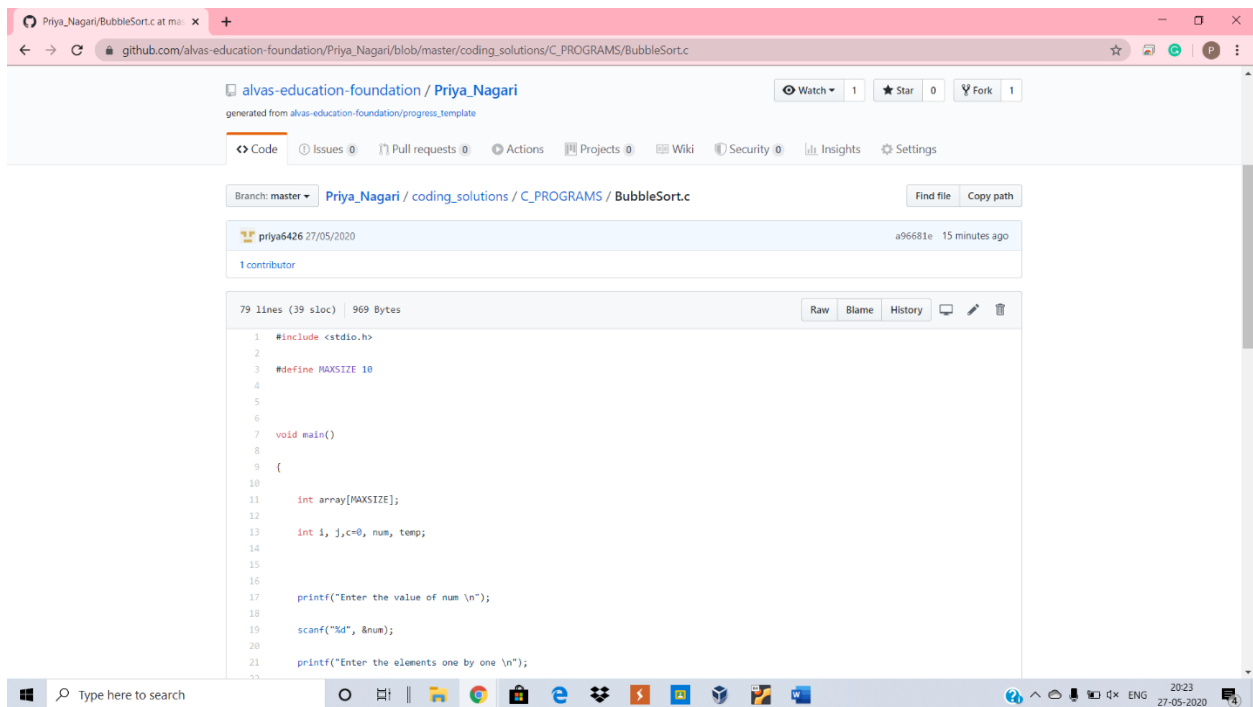
Certificate Provider: Udemy

total duration is 29 hours.

Today I learnt about activities life cycle and about fragments. And I completed previous classes challenge.

Online Coding Details:

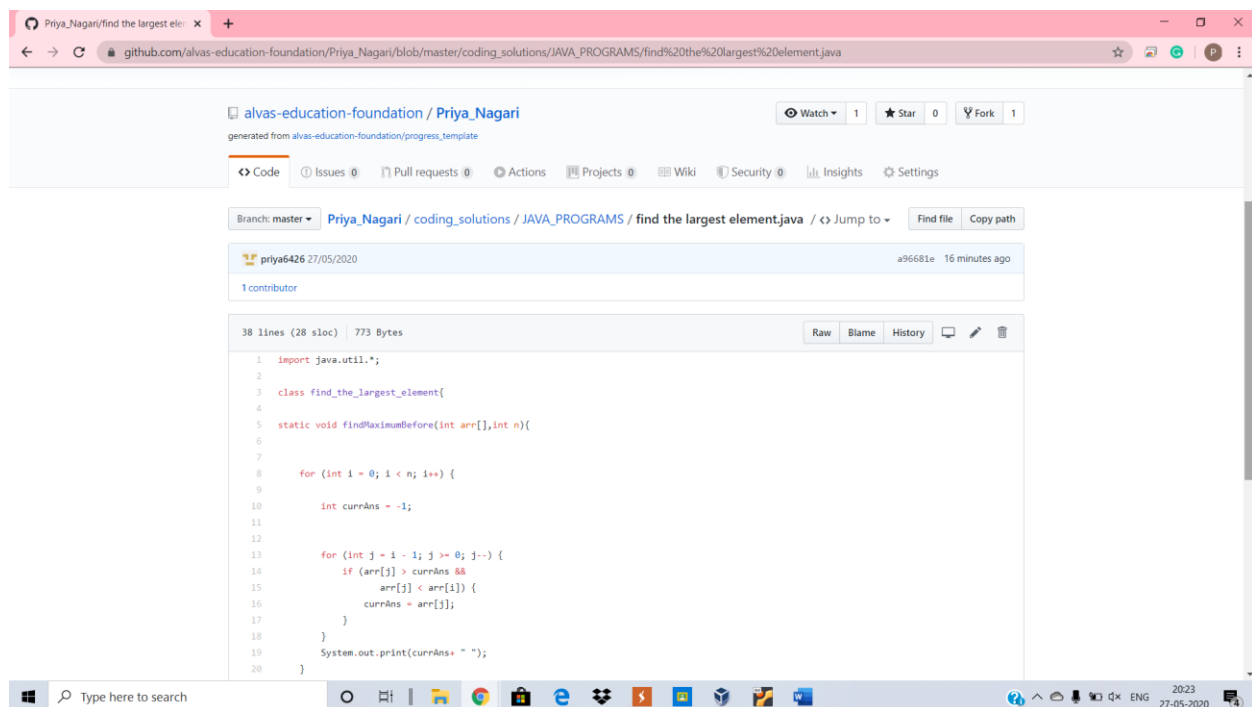
Problem Statement: 1. Problem Statement: 1. Write a C Program to sort an array of integers in ascending order and Display the sorted array and Number of passes performed for sorting.



The screenshot shows a web browser displaying a GitHub repository page for 'alvas-education-foundation / Priya_Nagari'. The repository is a fork of 'alvas-education-foundation/progress_template'. The file 'BubbleSort.c' is selected, showing its code. The code is a C program for bubble sort, with 79 lines and 969 bytes. The code includes standard headers, defines MAXSIZE as 10, and implements a main function that prompts the user for the number of elements and then for each element, before performing the sorting.

```
1 #include <stdio.h>
2
3 #define MAXSIZE 10
4
5
6
7 void main()
8
9 {
10
11     int array[MAXSIZE];
12
13     int i, j, c=0, num, temp;
14
15
16
17     printf("Enter the value of num \n");
18
19     scanf("%d", &num);
20
21     printf("Enter the elements one by one \n");
22
23 }
```

2. Given an array `arr[]` of the positive integers of size `N`, the task is to find the largest element on the left side of each index which is smaller than the element present at that index. Note: If no such element is found then print -1.



The screenshot shows a GitHub repository for the user 'Priya_Nagari'. The repository is named 'alvas-education-foundation / Priya_Nagari' and is generated from 'alvas-education-foundation/progress_template'. The file being viewed is 'find the largest element.java' located in the path 'coding_solutions / JAVA_PROGRAMS / find the largest element.java'. The file was committed by 'priya6426' on 27/05/2020. The code is a Java program that implements the task described in the problem statement. It uses a nested loop to find the largest element on the left side of each index which is smaller than the element present at that index. If no such element is found, it prints -1.

```
1 import java.util.*;
2
3 class find_the_largest_element{
4
5     static void findMaximumBefore(int arr[],int n){
6
7
8         for (int i = 0; i < n; i++) {
9
10             int currAns = -1;
11
12
13             for (int j = i - 1; j >= 0; j--) {
14                 if (arr[j] > currAns &&
15                     arr[j] < arr[i]) {
16                     currAns = arr[j];
17                 }
18             }
19             System.out.print(currAns+ " ");
20         }
21     }
22 }
```