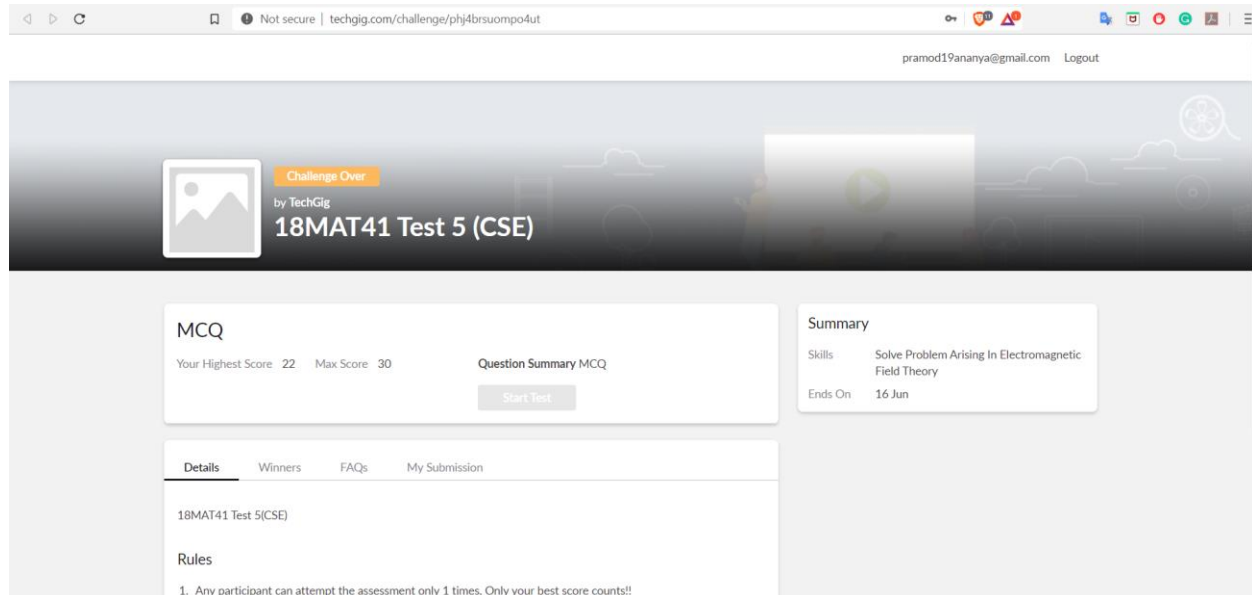


DAILY ONLINE ACTIVITIES SUMMARY

Date:	16/06/2020	Name:	Pramod R
Sem & Sec	4 th sem B section	USN:	4AL18CS059
Online Test Summary			
Subject	Complex Analysis probability and statistical methods		
Max. Marks	30	Score	22
Certification Course Summary			
Course	Java Programming for Complete Beginners		
Certificate Provider	Udemy	Duration	1 Hour
Coding Challenges			
Problem Statement: Write a python program to check whether a given a binary tree is a valid binary search tree (BST) or not.			
Status: Completed			
Uploaded the report in Github		YES	
If yes Repository name		https://github.com/alvas-education-foundation/Pramod_R	
Uploaded the report in slack		YES	

Online Test Details: (Attach the snapshot and briefly write the report for the same)



Complex Analysis probability and statistical methods was conducted. A total of 15 questions were there in which all the 15 of them were Multiple Choice Questions.

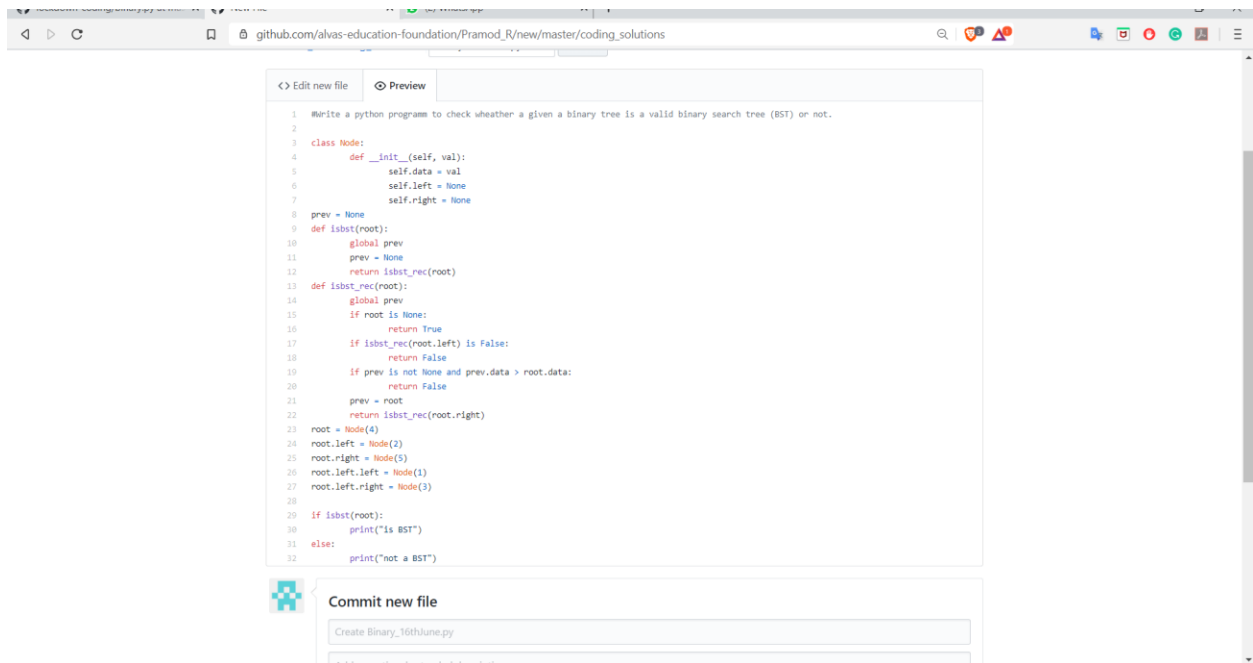
The above snapshot is the result sheet which was mailed to us by the Techgig team

Certification Course Details: (Attach the snapshot and briefly write the report for the same)

The screenshot shows the UdeMy website interface for the course 'Java Programming for Complete Beginners'. The main content area displays a video player with a thumbnail titled 'Multiplication Table - Step By Step' by 'in28minutes'. The thumbnail lists four steps: 'How to break it down?', 'Where do we start?', 'Calculate 5 * 5', and 'Print 5 * 5 = 25', followed by 'Do this 10 Times'. Below the video player, there is a section titled 'About this course' which states: 'Learn Java Programming with 200+ code examples. For Absolute Java Beginners! Start Learning Java Programming Now!'. On the right side, a 'Course content' sidebar lists 24 steps, each with a checkmark and a duration. The steps include: '17. Step 07 - Printing output to console with Java (12min)', '18. Step 08 - Printing output to console with Java - Exercise Statements (2min)', '19. Step 09 - Printing output to console with Java - Exercise Solutions (4min)', '20. Step 10 - Printing output to console with Java - Puzzles (8min)', '21. JShell Tip - Multiple Lines of Code (2min)', '22. Step 11 - Advanced Printing output to console with Java (7min)', '23. Step 12 - Advanced Printing output to console with Java - Exercises (4min)', and '24. Step 13 - Introduction to Variables in Java'.

The course I have chosen during the lockdown period is **Java Programming for Complete Beginners**. Since I had previously knew few topics about Java I am continuing this course. Since Java is used in major application development, I have chosen this course.

Coding Challenges Details: (Attach the snapshot and briefly write the report for the following)



The screenshot shows a GitHub web interface for a repository named 'alvas-education-foundation/Pramod_R/new/master/coding_solutions'. The main content is a Python file named 'Binary_16thJune.py'. The code defines a 'Node' class and a function 'isbst' to check if a binary tree is a valid Binary Search Tree (BST). The 'isbst' function uses a recursive helper 'isbst_rec' and a global 'prev' variable to ensure the in-order traversal of the tree is strictly increasing. The tree structure is defined with root node 4, left child 2, right child 5, and further children 1 and 3 under node 2.

```
1 #Write a python program to check wheather a given a binary tree is a valid binary search tree (BST) or not.
2
3 class Node:
4     def __init__(self, val):
5         self.data = val
6         self.left = None
7         self.right = None
8
9 prev = None
10 def isbst(root):
11     global prev
12     prev = None
13     return isbst_rec(root)
14 def isbst_rec(root):
15     global prev
16     if root is None:
17         return True
18     if isbst_rec(root.left) is False:
19         return False
20     if prev is not None and prev.data > root.data:
21         return False
22     prev = root
23     return isbst_rec(root.right)
24 root = Node(4)
25 root.left = Node(2)
26 root.right = Node(5)
27 root.left.left = Node(1)
28 root.left.right = Node(3)
29
30 if isbst(root):
31     print("Is BST")
32 else:
33     print("not a BST")
```

Below the code editor, there is a 'Commit new file' button and a text input field containing 'Create Binary_16thJune.py'.

The question I took to code is:

Write a python program to check whether a given a binary tree is a valid binary search tree (BST) or not.

Solution: The above snapshot is the code which I have uploaded in my Github repository