DAILY ONLINE ACTIVITIES SUMMARY

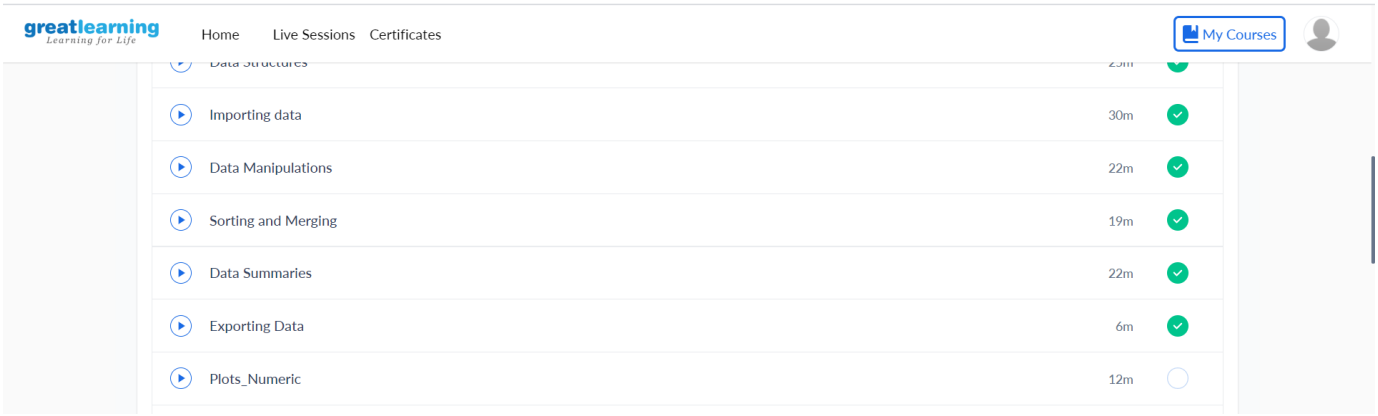
Date:	**12/06/2020**	**Name:**	**Raghavendra s**
Sem & Sec	**8 sem B sec**	**USN:**	**4AL16CS071**
Online Test Summary			
Subject	**BDA**		
Max. Marks	**60**	**Score**	**Not taken due to network problem**
Certification Course Summary			
Course	**Introduction to R**		
Certificate Provider	**greatlearning**	**Duration**	**3.00hrs**
Coding Challenges			
Problem Statement:			
Status: Solved			
Uploaded the report in Github		**Uploaded**	
If yes Repository name		**Raghavendra s**	
Uploaded the report in slack		**yes**	

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

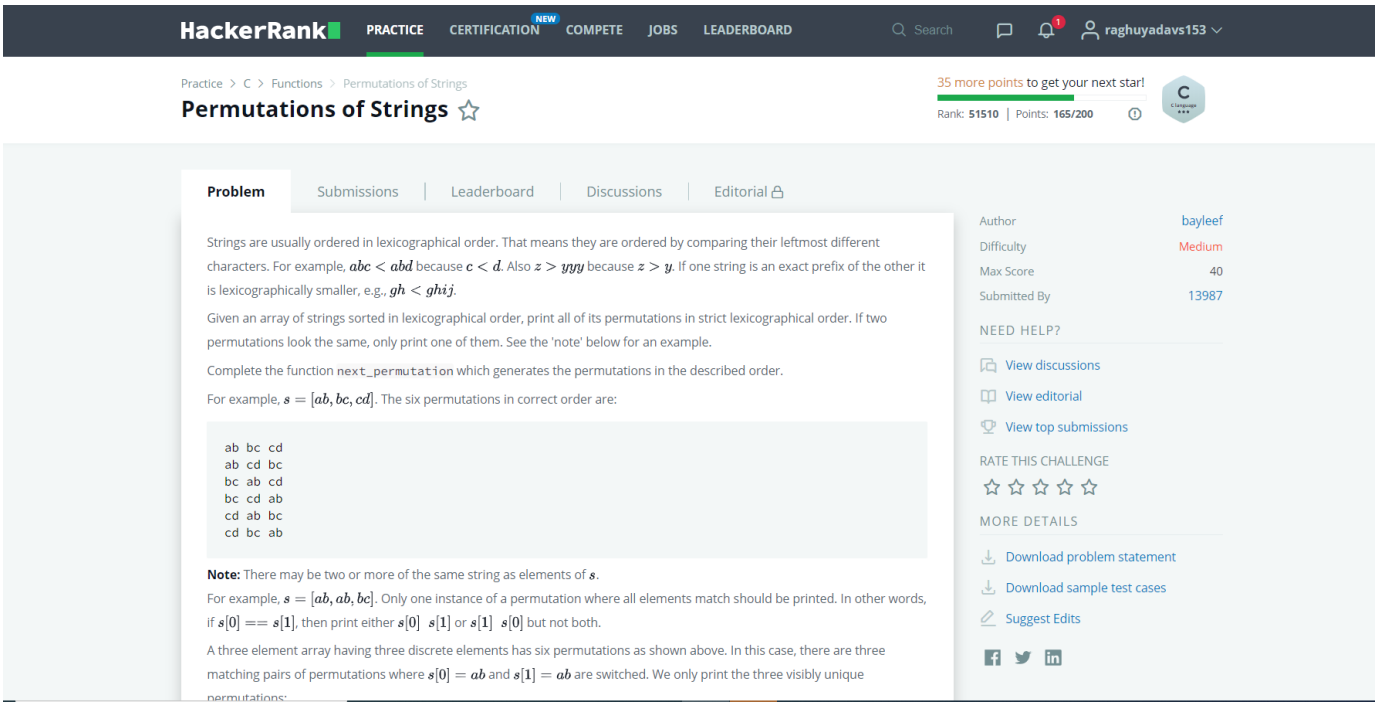
Certification Course Details: (Attach the snapshot and briefly write the report for the same)

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

online certificate



ONLINE CODDING



```
ab ab bc
ab bc ab
bc ab ab
```

Input Format

The first line of each test file contains a single integer n, the length of the string array s.

Each of the next n lines contains a string $s[i]$.

Constraints

- $2 \leq n \leq 9$
- $1 \leq |s[i]| \leq 10$
- $s[i]$ contains only lowercase English letters.

Output Format

Print each permutation as a list of space-separated strings on a single line.

Sample Input 0

```
2
ab
cd
```

Sample Output 0

```
ab cd
cd ab
```

Sample Input 1

Program:

```
int next_permutation(int n, char **s){
  // Find non-increasing suffix
  int i = n-1;
  while(i>0 && strcmp(s[i-1],s[i])>=0)
    i--;     // find key
  if (i<=0) return 0;

  // Swap key with its successor in suffix
  int j = n-1;
  while(strcmp(s[i-1],s[j])>=0)
    j--;     // find rightmost successor to key
  char *tmp = s[i-1];
  s[i-1] = s[j];
  s[j] = tmp;
```

```
    // Reverse the suffix
    j = n-1;
    while(i<j) {
        tmp = s[i];
        s[i] = s[j];
        s[j] = tmp;
        i++;
        j--;
    }
    return 1;
}
```