**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Date:** | | **11/06/2020** | **Name:** | **JASLINE SHARON TAURO** | |
| **Sem & Sec** | | **4th sem, A Section** | **USN:** | **4AL18CS029** | |
| **Online Test Summary** | | | | | |
| **Subject** | **------** | | | | |
| **Max. Marks** | **------** | | **Score** | **------** | |
| **Certification Course Summary** | | | | | |
| **Course** | | 1. [**INTRODUCTION TO INFORMATION SECURITY**](https://olympus.greatlearning.in/courses/11264) | | | |
| **Certificate Provider** | | **Great Learning Academy** | **Duration:** | | **3 HRS** |
| **Coding Challenges** | | | | | |
| **Problem Statement:**   1. Write a Java Program to Segregate Even and Odd numbers. | | | | | |
| **Status: EXECUTED** | | | | | |
| **Uploaded the report in GitHub** | | | **YES** | | |
| **If yes Repository name** | | | <https://github.com/jaslinesharontauro/JAVA_Prgms> | | |
| **Uploaded the report in slack** | | | **YES** | | |

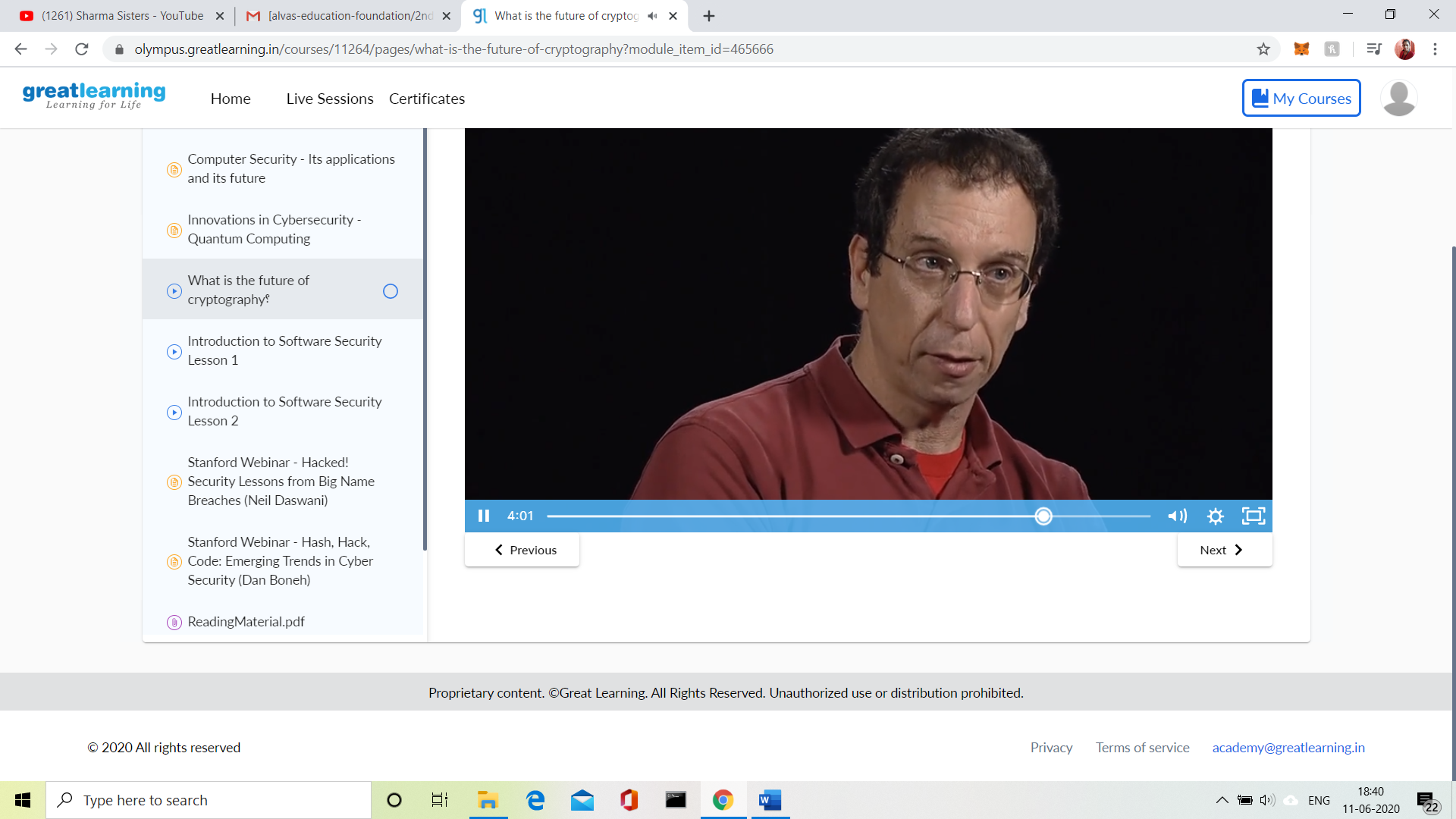
**Online Test Details: (Attach the snapshot and briefly write the report for the same)**

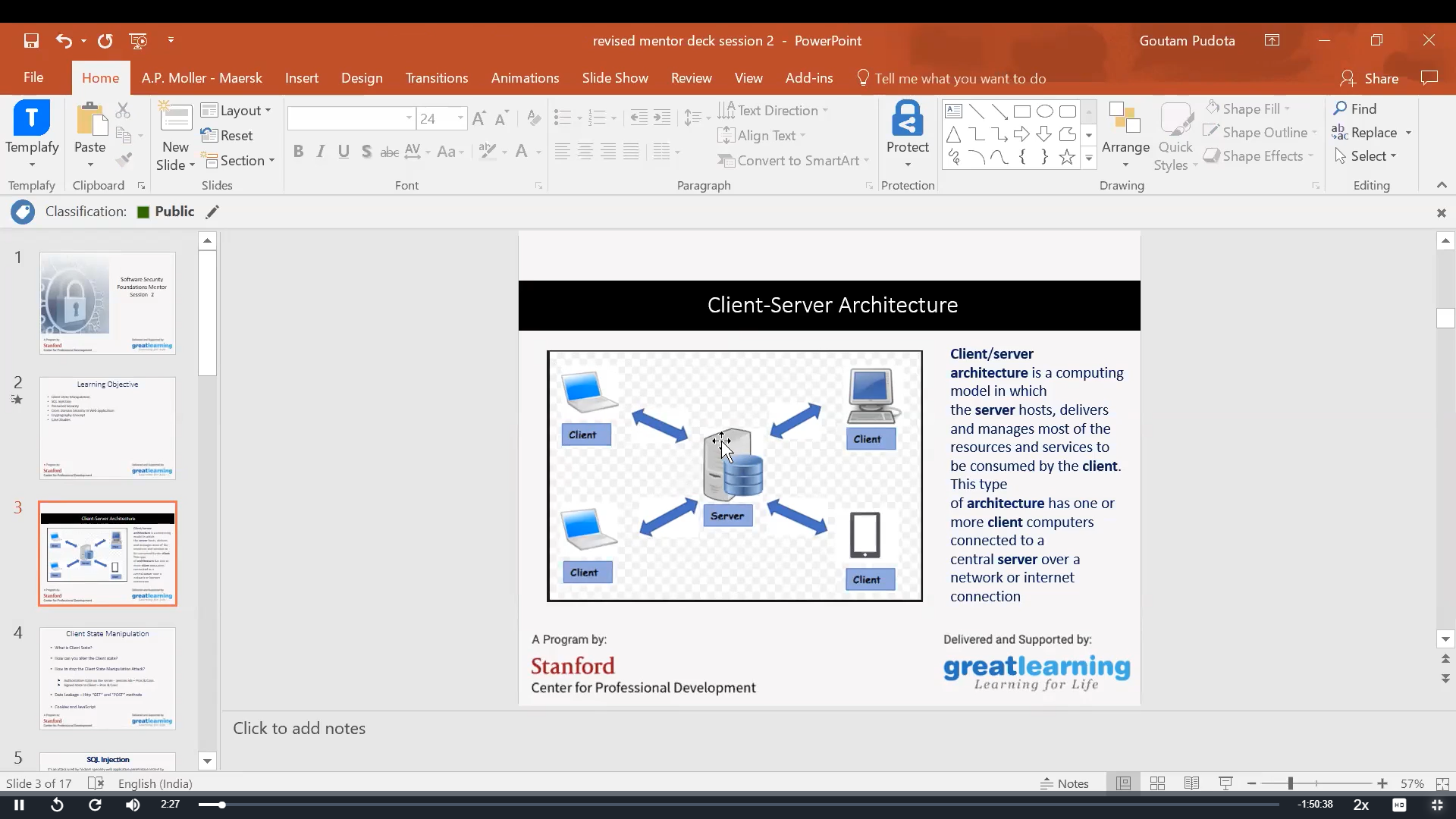
**Certification Course Details: (Attach the snapshot and briefly write the report for the same)**

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

**1.CERTIFICATION COURSE DETAILS**

**Today I have done the Course** Introduction to Software Security. I **learnt about cryptography and its future advances and also learnt about cryptography in quantum computing.**





**3.CODING CHALLENGES DETAILS:**

Problem Statement 1:

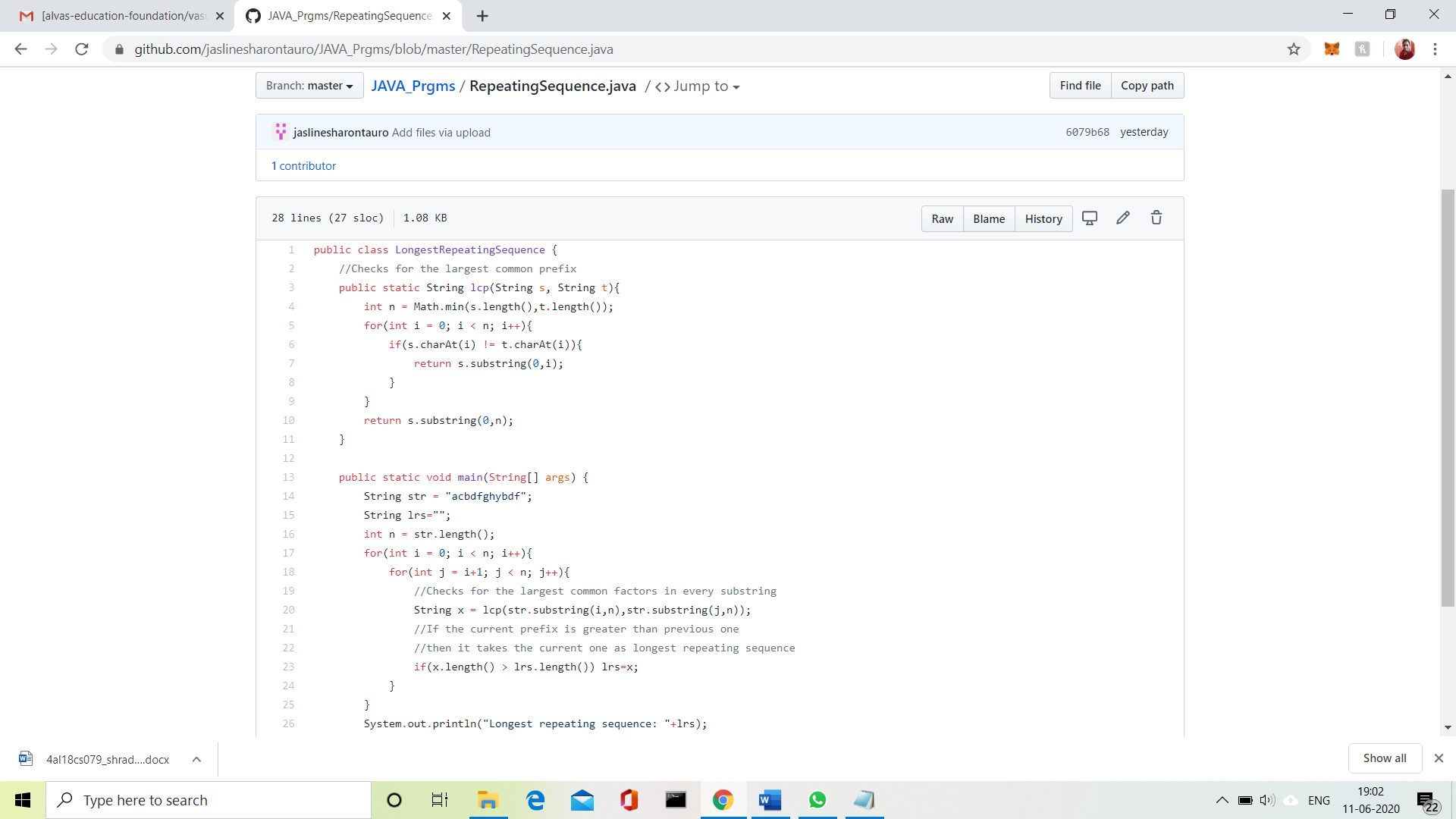
Write a Java Program to Segregate Even and Odd numbers

Given an array A [], write a function that segregates even and odd numbers. The functions should put all even numbers first, and then odd numbers.  
Example:

Input = {12, 34, 45, 9, 8, 90, 3}  
Output = {12, 34, 8, 90, 45, 9, 3}

Algorithm: segregateEvenOdd ()

1. Initialize two index variables left and right:  
   left = 0, right = size -1
2. Keep incrementing left index until we see an odd number.
3. Keep decrementing right index until we see an even number.
4. If lef < right then swap arr[left] and arr[right]



**Solution Uploaded in GitHub.**