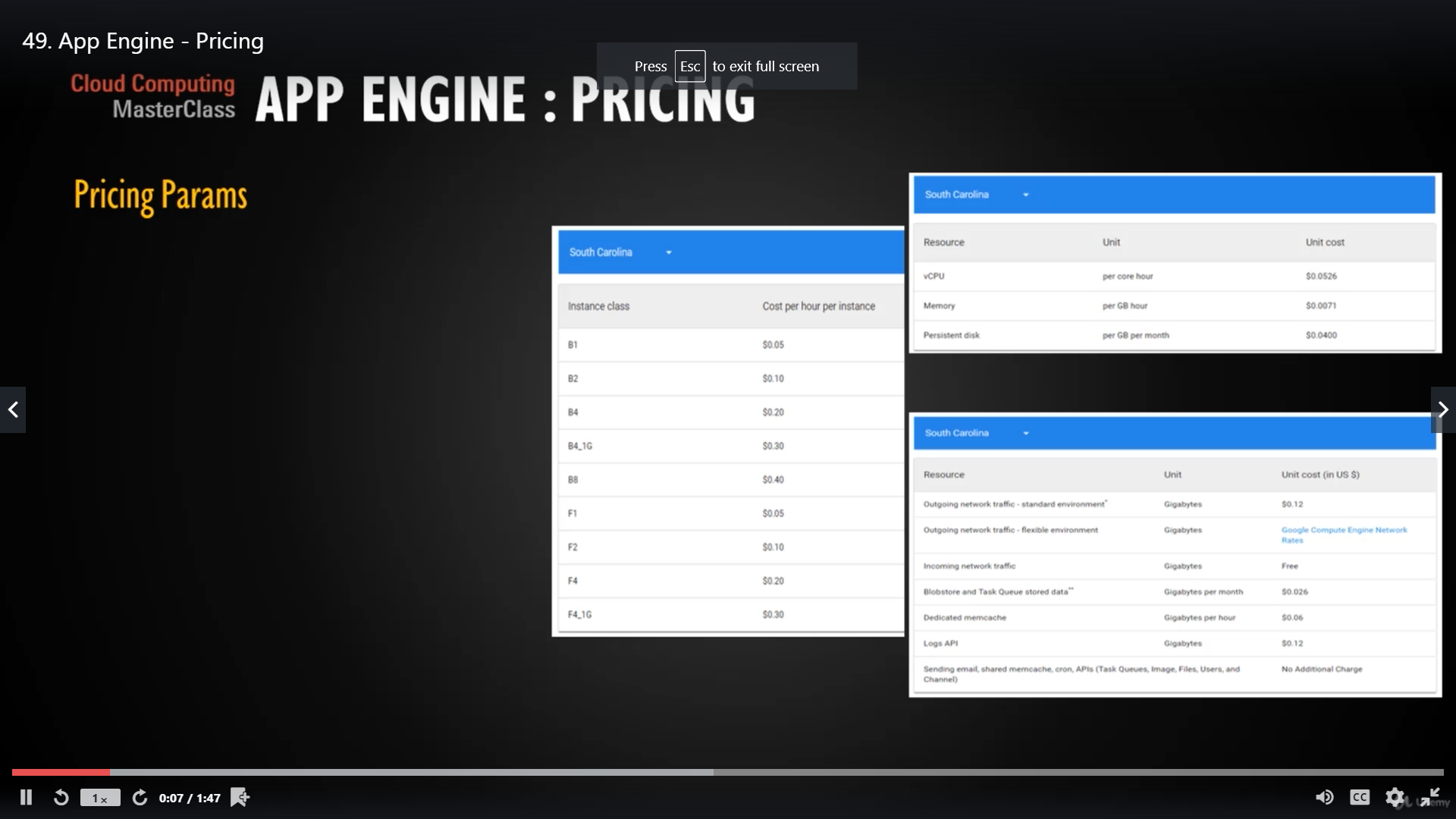
**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **11/06/2020** | | | | | **Name:** | **RACHANA B S** | |
| **Sem & Sec** | **4th Sem B Sec** | | | | | **USN:** | **4AL18CS065** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **Not conducted** | | | | | | |
| **Max. Marks** | | **-** | | **Score** | | | **-** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Introduction to Google Cloud** | | | | | | | |
| **Certificate Provider** | | | **Udemy** | | **Duration** | | | **2 weeks** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:1.** [Write a Java Program to Segregate Even and Odd numbers](https://github.com/orgs/alvas-education-foundation/teams/2nd-year/discussions/114)  Top of Form  Bottom of Form  2. Java Program to find the longest repeating sequence in a string. | | | | | | | | |
| **Status: executed** | | | | | | | | |
| **Uploaded the report in GitHub** | | | | | **yes** | | | |
| **If yes Repository name** | | | | | <https://github.com/bsrachana/lockdown_coding> | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

Online Test Details: test was not conducted today.

Certification Course Details:

In today’s session, I learnt about App Engine Pricing and Pricing Params.



Coding Challenges Details:

Every day we are given with new question of coding related to the language of java and c. it seems interesting how we imbibe ourselves in depth to understand the logic, break it and then code for it.

Today’s question was:

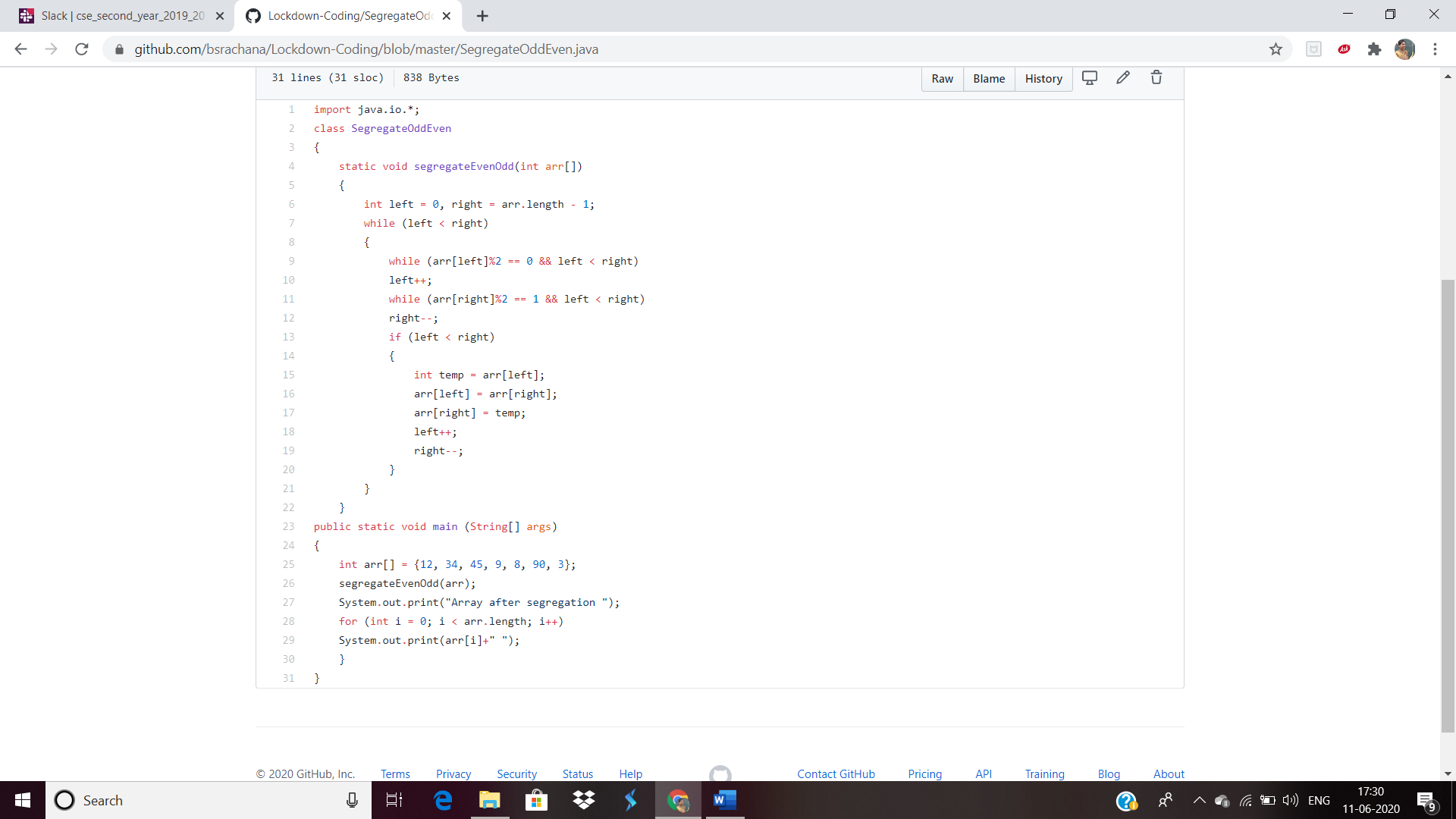
1. [Write a Java Program to Segregate Even and Odd numbers](https://github.com/orgs/alvas-education-foundation/teams/2nd-year/discussions/114)

Top of Form

Bottom of Form

|  |
| --- |
| 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] |

Snapshot:



2. Java Program to find the longest repeating sequence in a string  
string:acbdfghybdf

SNAPSHOT:

