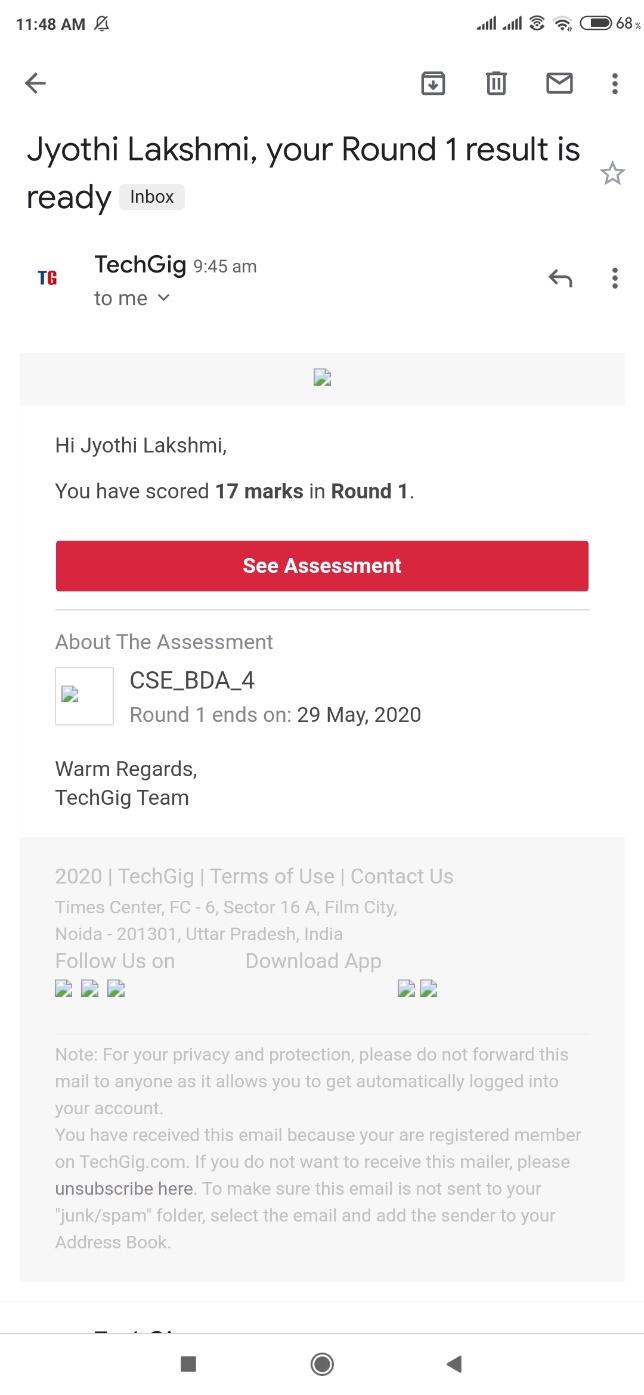
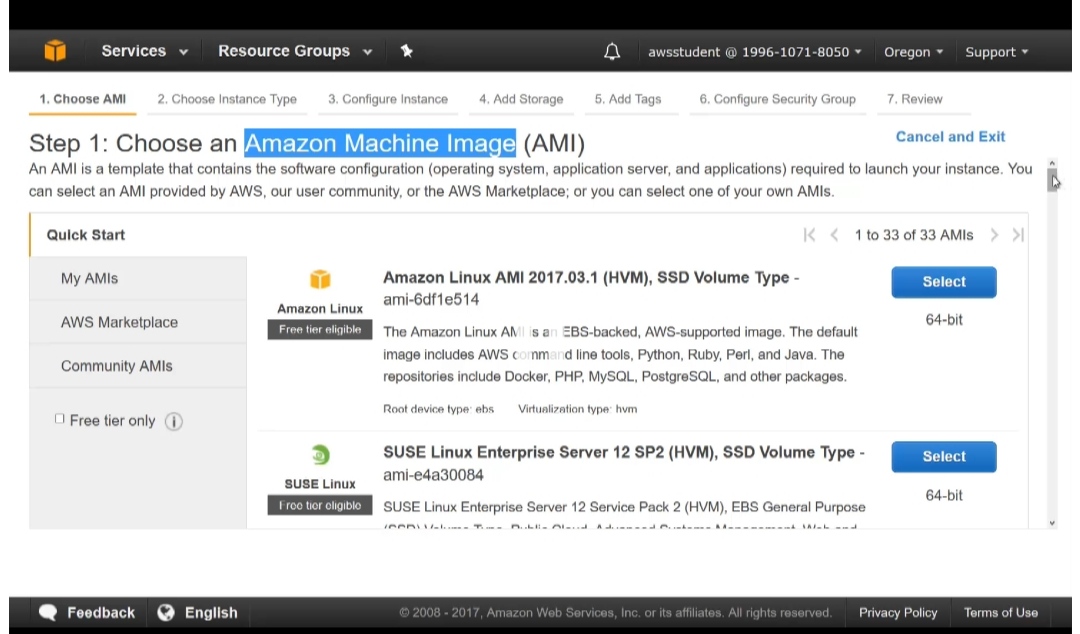
**D AILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **29/5/2020** | | | | **Name:** | **Jyothi Lakshmi** | |
| **Sem & Sec** | **8th Sem** | | | | **USN:** | **4AL16CS129** | |
| Online Test Summary | | | | | | | |
| **Subject** | | **Big Data Analysis** | | | | | |
| **Max. Marks** | | **30** | | **Score** | | **17** | |
| Certification Course Summary | | | | | | | |
| **Course** | **The Introduction to Amazon Elastic Compute Cloud(EC2)** | | | | | | |
| **Certificate Provider** | | | **AWS** | **Duration** | | | **10min** |
| Coding Challenges | | | | | | | |
| **Problem Statement:**  1: Given an arayar[] of size N and an integer K.The task is to find the last remaining element in the aray after reducing the aray. | | | | | | | |
| **Status:COMPLETED** | | | | | | | |
| **Uploaded the report in Github** | | | | **Yes** | | | |
| **If yes Repository name** | | | | **Jyothi\_129** | | | |
| **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)

Program 1.

Void more Than N dK(intar[],intn,intk) {

/kmustbegreaterthan1togetsomeoutput if(k<2) return;

/Step1:Create a temporary aray(contains element and count) of size k-1. Initializecountofal elementsas0/

Struct eleCounttemp[k-1]; for(inti=0;i<k-1;i++) temp[i].c=0;

/Step2:Process an elements of input aray/ for(inti=0;i<n;i++)

{

intj;

/If ar[i] is already present in the element count aray, then increment its count/ for(j=0;j<k-1;j++) {

if(temp[j].e==ar[i]) { temp[j].c+=1; break; }

}

/Ifar[i] is not present in temp[]/ if(j==k-1)

{

intl;

/If there is position available in temp[], then place ar[i] in the first available position and set count as 1/

for(l=0;l<k-1;l++) {

if(temp[l].c==0) {

temp[l].e=ar[i]; temp[l].c=1; break;

} }

/If all the position in the temp[] are filed, then decrease count of every element by 1/ if(l==k-1) for(l=0;l<k;l++)

temp[l].c-=1; }

}

/Step3:Check actual counts of potential candidates in temp[]/ for(inti=0;i<k-1;i++)

{

/Calculate actual count of elements in tac=0;/ actual count for(int j=0; j<n; j++) if(ar[j]==temp[i].e) ac++;

/If actual count is more than n/k, then print it if(ac>n/k) cout<<"Number:"<<temp[i].e <<"Count:"<<ac<<endl;

}