**DAILY ONLINE ACTIVITIES SUMMARY**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Date:** | **29/5/2020** | | | | | **Name:** | **Amogha U** | |
| **Sem & Sec** | **8th Sem** | | | | | **USN:** | **4AL16CS010** | |
| **Online Test Summary** | | | | | | | | |
| **Subject** | | **Big Data Analytics** | | | | | | |
| **Max. Marks** | | **30** | | **Score** | | | **19** | |
| **Certification Course Summary** | | | | | | | | |
| **Course** | **Introduction to information security** | | | | | | | |
| **Certificate Provider** | | | **greatlearning** | | **Duration** | | | **5.5hrs** |
| **Coding Challenges** | | | | | | | | |
| **Problem Statement:**  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** | | | | | **amogha-u** | | | |
| **Uploaded the report in slack** | | | | | **yes** | | | |

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

A screenshot of a cell phone

Description automatically generated

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

A screenshot of a computer

Description automatically generated

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

**Program 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.

|  |  |
| --- | --- |
|  |  |
|  | Void more Than N dK(intar[],intn,intk) { |
|  | /k must be greater than 1 to get some output 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; |
|  | } |