Name: Roll Number:

# ESO207: Data Structures and Algorithms

Programming Assignment 2

Due Date: 19th Feb, 2022

Total Number of Pages: 3 Total Points 100

** Submit all the codes on Moodle*

**Question 1. Magical Rods**

(20 Points) There are $N$ magical cylindrical rods, each having its own length. Consider two rods $R_1$ and $R_2$, rod $R_1$ can cover rod $R_2$ if rod $R_1$ is atleast two times greater(in length) than the length of rod $R_2$. Also each rod can cover atmost one rod, and if a rod is covered by some other rod then this rod cannot cover any rod. Design an algorithm that outputs the count of the *visible* rods such that minimum no. of rods are *visible*.

*Note: A rod is said to be visible if it is not covered by any rod.*

You will be given an array $A$ of size $N$. The values in the array $A$ represents the length of the rods.

**Input Format:**

The first line of each test case contains an integers $N$. The second line of each test case contain $N$ integers which represents the array $A$.

**Constraints:**

$1 \leq N \leq 10^5$

$1 \leq A_i \leq 10^6, \forall i : 0 \leq i < N$

**Output Format:**

Print the count of the visible rods such that minimum no. of rods are visible.

**Example:**

Let $N = 8$ and $A = [9, 1, 6, 2, 6, 5, 8, 3]$.
The output should be 5.

**Explanation:**

One of the possible arrangement to get the optimal answer can be:
Rod with length 1 covers rod with length 5.
Rod with length 2 covers inside rod with length 8.
Rod with length 3 covers inside box with length 9.
Now no rod can cover other rod. Hence we will get 5 as the optimal answer.

**Question 2. Monkey Jump**

(20 points) There are $N$ buildings present in a street numbered 0 to $N-1$ in sequence. A monkey starts jumping from one of the buildings as the start point, with the following jumping conditions:-

- The monkey can only jump in the backwards direction and only on a consecutive building, i.e, from $i$ to $i-1$ and no other building.

- He can jump only when the height of the building he is about to jump upon is less than or equal to the building he started from.

Consider an array $K$ wherein index $i$ stores the maximum no. of buildings he can jump upon when building $i$ is taken as the starting point.
You will be given an array $A$ of size $N$ where element $i$ in the array represents the height of the building $i$. Design an algorithm that outputs the array $K$.

**Input Format:**

The first line of each test case contains an integers $N$. The second line of each test case contain $N$ integers which represents the array $A$.

**Constraints:**

$1 \leq N \leq 10^5$

$1 \leq A_i \leq 10^6, \forall i : 0 \leq i < N$

**Output Format:**

Print $N$ integers, separated by a space where $i^{th}$ integer is the $i^{th}$ element of the array K.

**Example:**
Let $N = 8$ and $A = [10, 8, 9, 5, 4, 6, 7, 9]$.
The output should be $[0, 0, 1, 0, 0, 2, 3, 6]$.

**Explanation:**
There are 8 buildings of heights $[10, 8, 9, 5, 4, 6, 7, 9]$. For building 0, Monkey cannot jump backwards. For building 2 , he can jump on the building 1, so answer for building 2 will be 1 and so on. Lastly, from building 7 , he can jump to all the previous buildings except building 0. So, answer for building 7 will be 6.

**Question 3. Special Count**

(30 points) Given an array $A$ of size N and an integer K. Let $B$ be an array where element at index $i$ in array $B$ represents the total no. of elements in array A which satisfy $A[i] > K \cdot A[j]$ for all $j > i$.

Design an algorithm that outputs the array $B$.

**Input Format:**

The first line of each test case contains two integers $N$ and $K$. The second line of each test case contain $N$ integers which represents the array $A$.

**Constraints:**

$1 \leq N \leq 10^5$

$1 \leq K \leq 10^5$

$1 \leq A_i \leq 10^6, \forall i : 0 \leq i < N$

**Output Format:**

Print $N$ integers all separated by a space where $i^{th}$ integer is the $i^{th}$ element of the array $B$.

**Example :**
Let $N = 4$, $K = 2$ and $A = [5, 2, 4, 1]$.
The output should be $[2, 0, 1, 0]$.

**Explanation :**
There are 2 elements such that $A[0] > 2 \cdot A[j]$ for all $j > 0$.
There are 0 elements such that $A[1] > 2 \cdot A[j]$ for all $j > 1$.
There are 1 elements such that $A[2] > 2 \cdot A[j]$ for all $j > 2$.

There are 0 elements such that $A[3] > 2 \cdot A[j]$ for all $j > 3$.

**Question 4. School Trip**

(30 points) The class teacher is planning to take her students on a trip to Goa. She wants to book rooms in a hotel, the hotel consists of $N$ rooms in which some of them are occupied. There are total $M$ students who are willing to go for the trip, she wants to book total $M+1$ rooms (including herself) such that the maximum distance from any student to her is minimised. Design an algorithm that outputs the minimum possible distance of the student farthest from her.

You will be given an array $A$ of size $N$. Each index will have values 0 or 1, 0 represents the room is unoccupied and 1 represents that room is occupied. The distance between two rooms is defined as the absolute difference between the indices of the room.

**Input Format:**

The first line of each test case contains two integers $N$ and $M$. The second line of each test case contain $N$ integers which represents the array $A$.

**Constraints:**

$1 \leq N \leq 10^5$

$1 \leq M \leq 10^5$, $M + 1 \leq N$

$0 \leq A_i \leq 1, \forall i : 1 \leq i \leq N$

**Output Format:**

Print the minimum possible distance of the student farthest from the teacher.

**Example :**
Let $N = 7$, $m = 2$ and $A = [0, 1, 0, 0, 1, 0, 0]$.
The output should be 2.

**Explanation :**
Teacher can book room 3 for herself, rooms 1 and 4 for students. The maximum distance will be 2.

**Note :**

- All questions have to be answered through a contest in Hackerrank. The contest has 4 challenges, each corresponding to a part. You have to submit your code through the contest. Following is the link to the contest: `https://www.hackerrank.com/eso207a-pa2-2021-22-ii`
- Your codes will be checked for possible plagiarism of any sorts. If we find such cases, then we will possibly award an F grade.
- Time limit for all the questions is 1 second
- Allowed Languages for challenge code submission : C, C++
- Allowed libraries : stdio.h for C and iostream for C++
- You will also need to upload all your program files (C/C++) on moodle.