

ADSA-2021 Lab Assignment - 2 Duration: 3 Hrs 30 minutes (Time:2PM to 5.30PM)

INSTRUCTIONS

1. ALL FOUR PROBLEMS are COMPULSORY
2. Carefully read all assignment problems.
3. Write only a single main function. You can call the required functions from the main function. Print the list of elements wherever necessary.
4. Name the file as follows: **S2020xxxxx_A02.c**
5. DO NOT zip. Upload a single .c file directly to your submission in the common Google classroom.
6. **Don't share or copy the codes. If malpractice is found, you will be awarded Zero.**

QUESTION 1:

2-Points

Rohan is at the lottery shop where he wants to buy lottery tickets. He checked the lottery tickets and found that there are N lottery tickets in that shop. Each lottery ticket has some score value, ideally it is like lottery tickets which have a higher price, have a higher score value and no two lottery tickets in the shop have the same price.

He is very excited by looking at these lottery tickets. He decided to buy K lottery tickets while maximizing their score value. Could you please help Rohan to determine the exact money he needs to buy the lottery tickets.

(Note that: while solving a problem you must consider a merge sort algorithm).

Input Format

- First line of input contains two space separated integers N and Q, where N is the number of lottery tickets available and Q is the number of queries.
- Next line contains N space separated integers denoting the price T_1 of N tickets.
- Next line contains N space separated integers denoting the scores S_i of N tickets.
- Next Q lines contain a number K each, the number of tickets she wants to purchase.

Output Format

Print the money required for each query in a separate line

Constraints

$$1 \leq N \leq 10^5$$

$$1 \leq Q \leq 10^5$$

$$1 \leq K \leq N$$

$$1 \leq T_1, S_i \leq 10^9$$

Sample Input	Sample Output
5 2 2 3 9 4 5 3 5 11 6 7 5 3	23 18

Question-2

3-points

- Write a program to perform following operations on stack.
- a) Create functions for push and pop operations of stack.
- b) Write a function to convert an **infix expression to a postfix expression**. Pass a one dimensional character array P to the function as input (infix exp) and return character array Q (postfix exp). Test your program for following input
 P : (A – (B / C) * D + E) * F % G
- c) Write a function for the evaluation of a given postfix expression. For testing pass the postfix expression Q of part b and supply the following set of values.
- A = 90, B = 50, C = 2, D = 3, E = 1, F = 2, G = 5

Question-3

2-points

Implement the randomized quick sort algorithm and identify the complexity of the program

Question-4

3-points

Given an integer array, move all zeros present in it to the end. The solution should maintain the relative order of items in the array.

Input: { 6, 0, 8, 2, 3, 0, 4, 0, 1 }

Output: { 6, 8, 2, 3, 4, 1, 0, 0, 0 }