DSA Final Lab exam

Section A

July 5, 2022

Instructions for the exam

1. Place your Permanent / Temporary Student ID card on the desk during the examination for verification by the Invigilator.
2. Reading material such as books are not allowed inside the examination hall.
3. Borrowing writing material or calculators from other students in the examination hall is prohibited.
4. If any student is found indulging in malpractice or copying in the examination hall, the student will be given 'F' grade for the course and may be debarred from writing other examinations.
5. This is a 4 hour exam.
6. You will be submitting your problems on the OJ
7. All the data structures will be provided in a .c file. You can copy them into your code and use them as required. You can modify them as well. Please go through the template code before using it (at-least the function definitions).
8. Refer to the time limit and memory limit of each problem on OJ.
9. There are 4 problems.
    - **Problem A:** 100 points
    - **Problem B:** 200 points
    - **Problem C:** 200 points
    - **Problem D:** 300 points

Problem A: HTML Compiler (100 points)

Kunwar is making a parser for HTML files. An HTML file has different tags surrounded by the angular brackets < and >.

The first thing he needs to do for this parser is find the longest prefix of the HTML file that forms a valid bracket sequence.

To make his job easier, he has removed all the other characters from the file and now has a string of length N with just the angular brackets. Can you help him find the longest prefix of this string that forms a valid bracket sequence.

Input format

The first line contains T, the number of test cases.

T lines follow, each containing a string that Kunwar needs to parse.

Output format

For each testcase, output the length of the longest prefix that makes a valid bracket sequence on a separate line.

Constraints

$T \leq 10$
$N \leq 10^5$

Subtasks

Subtask 1(5 points): $N \leq 10$
Subtsk 2(5 points): $N \leq 100$
Subtask 3(5 points): $N \leq 1000$
Subtask 4(85 points): Original constraints

Examples

Input 1

4
<<>>
><
<>>>
><><><><>

Output 1

4
0
2
0

Explanation

For first testcase, the entire string is a valid bracket sequence.
For the second one, there is no prefix with a valid bracket sequence.
For the third one, <> is a valid bracket sequence but all the prefix longer than that do not form a valid sequence
For the last one, there is no prefix that is valid. Note that the substring from the second character to the last one does form a valid sequence. However, it is not a prefix.

Problem B: The Smallest String Concatenation (200 points)

You're given n strings $a_1, a_2, \ldots, a_n$. You can concatenate them together in any order. **Note that you have concatenate all of them to get the resulting string.**

String x is lexicographically smaller than string y if either x is a prefix of y and $x \neq y$, or if there exists an index i $(1 \leq i \leq \min(|x|, |y|))$ such that $x_i < y_i$ and for every j $(1 \leq j < i)$ $x_j = y_j$. Here $|x|$ denotes the length of string x.

Given the list of strings, output the resulting concatenated string that is lexicographically smallest.

Constraints

$1 \leq n \leq 50000$
$1 \leq |a_i| \leq 50$ For all $i = 1, \ldots, n$
All strings a_i consist of lowercase latin alphabets ONLY.

Input format

First line contains an integer n.
Next n lines contain the n strings $a_1, a_2, \ldots, a_n$, one on each line. All strings a_i consist of lowercase latin alphabets ONLY.

Output format

Resulting concatenated string that is lexicographically smallest.

Examples

Sample Testcase 1

Input

4
abba
abacaba
bcd
er

Output

abacabaabbabcder

Explanation

You concatenate the 2^{nd}, 1^{st}, 3^{rd} and 4^{th} strings in this particular order to get the lexicographically smallest string.

Sample Testcase 2

Input

5
x
xx
xxa
xxaa
xxaaa

Output

xxaaaxxaaxxaxxx

Explanation

You concatenate the 5^{th}, 4^{th}, 3^{rd}, 2^{nd} and 1^{st} strings in this particular order to get the lexicographically smallest string. You can also concatenate the 5^{th}, 4^{th}, 3^{rd}, 1^{st} and 2^{nd} strings to get the same resulting string.

Sample Testcase 3

Input

3
c
cb
cba

Output

cbacbc

Explanation

You concatenate the 3^{rd}, 2^{nd} and 1^{st} strings in this particular order to get the lexicographically smallest string.

Problem C (200 points)

You can safely skip the following paragraph.

Your TA was asked to come up with a problem for the lab exam. He "referred" to an online resource, "discussed the logic" with his friends and came up with the following novel problem. Profs are convinced that he has copied the problem. He claims it's just a coincidence. You be the judge and help your profs arrive at the truth.

Suppose LeetCodeforces will start its IPO soon. In order to sell a good price of its shares to Venture Capital, LeetCodeforces would like to work on some projects to increase its capital before the IPO. Since it has limited resources, it can only finish at most k distinct projects before the IPO. Help LeetCodeforces design the best way to maximize its total capital after finishing at most k distinct projects.

You are given n projects where the i^{th} project has a pure profit $profits[i]$ and a minimum capital of $capital[i]$ is needed to start it.

Initially, you have w capital. When you finish a project, you will obtain its pure profit and the profit will be added to your total capital.

Pick a list of at most k distinct projects from given projects to maximize your final capital, and return the final maximized capital.

The answer is guaranteed to fit in a 32-bit signed integer.

Constraints

$1 \leq k \leq 10^5$
$0 \leq w \leq 10^9$
$n == profits.length$
$n == capital.length$
$1 \leq n \leq 10^5$
$0 \leq profits[i] \leq 10^4$
$0 \leq capital[i] \leq 10^9$

Subtasks

Subtask 1(10 points): $n \leq 10$
Subtsk 2(20 points): $n \leq 10^3$
Subtask 3(170 points): Original constraints

Input format

First line contains three integers n, k, and w.
Next n lines contain a pair of space-separated integers $profits[i]$ and $capital[i]$, on each line.

Output format

The final maximized capital.

Examples

Sample Testcase 1

Input

3 2 0
1 0
2 1
3 1

Output

4

Explanation

Since your initial capital is 0, you can only start the project indexed 0. After finishing it you will obtain profit 1 and your capital becomes 1. With capital 1, you can either start the project indexed 1 or the project indexed 2. Since you can choose at most 2 projects, you need to finish the project indexed 2 to get the maximum capital. Therefore, output the final maximized capital, which is $0 + 1 + 3 = 4$.

Sample Testcase 2

Input

3 3 0
1 0
2 1
3 2

Output

6

Problem D: Duathlon (300 points)

PEC is organising a duathlon on a sunny sunday morning for 3 attendance. The race track for the duathlon has N (1 indexed) checkpoints and M roads connecting those checkpoints. Each road is undirected and connects 2 checkpoints. For each road you know the time taken to traverse it on foot and the time taken to traverse it on a cycle. At each node there is a facility to pickup a cycle, the time taken to pick up a cycle at any particular node is different and given to you. All the nodes form a connected graph.

For each graph you will be given up to 10 queries, where you are given the start node (u) and the end node (v) and you are required to find the shortest time to go from u to v. You must rent out a cycle on any node on your path to the destination (You are allowed to acquire the cycle on either of the start or the end node. However, you are not allowed to leave the cycle once you've taken it on any node apart from v). It is mandatory for you to take up a cycle once during the entire course (it can be on the last node as well).

Input format

The first line of the input contains 2 integers, N and M.

The second line of the input contains the array C of N integers where C_i is the time taken to acquire the cycle at node i.

M lines follow, containing 4 integers, u, v, w and c. This denotes that there is a road between nodes u and v on which it takes you w seconds to walk and c seconds to cycle.

The next line contains a single integer, Q, the number of queries.

Q lines follow with 2 integers, U and V denoting start node U for the track and end node V.

Output format

For each query, output a single integer on a separate line, the minimum amount of time to go from the start to the end node.

Constraints

$2 \leq N \leq 10^5$
$N - 1 \leq M \leq \min(10^6, \frac{N*(N-1)}{2})$
$0 \leq C_i, w, c \leq 10^9$
$Q \leq 10$

Subtasks

Subtask 1(15 points): $N \leq 10$
Subtask 2(45 points): $N \leq 1000$
Subtask 3(30 points): $M = N - 1$
Subtask 4(15 points): $C_i = 0$
Subtask 5(30 points): $w = c$
Subtask 6(165 points): Original constraints

Examples

Sample Testcase 1

Input

5 6
3 1 4 5 2
1 2 2 3
2 3 3 1
3 4 6 6
4 5 2 8
2 4 7 4
1 5 2 3
5
1 4
1 3
2 5
5 3
4 5

Output

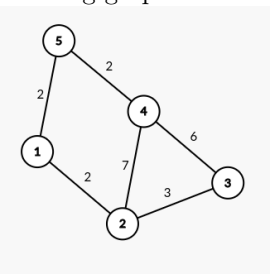
7
4
6
6
4

Explanation

Path for first query:
Walk from 1 to 2 [2 time units]
Get a cycle at 2 [1 time unit]
Cycle from 2 to 4 [4 time units]

Total time taken = 7 units

Walking graph:



Cycling graph: