2020 Team Formation Test (Round 2)

Task Overview

ID	Name	Time Limit	Memory Limit	Subtasks
T201	Font Size	1.000 s	256 MB	3 + 5 + 10 + 10 + 25 + 47
T202	Let's Lift String Swiftly	1.000 s	256 MB	17 + 15 + 25 + 43
T203	Entropy Maximizer	1.000 s	256 MB	15 + 9 + 11 + 23 + 42
T204	Metros of Runeterra	Output-Only Task		

Notice:
Unless otherwise specified, inputs and outputs shall follow the format below:

- One space between a number and another number or character in the same line.
- No space between characters in the same line.
- Each string shall be placed in its own separate line.
- Outputs will be automatically fixed as follows: Trailing spaces in each line will be removed and an end-of-line character will be added to the end of the output if not present. All other format errors will not be fixed.

C++ programmers should be aware that using C++ streams (`cin` / `cout`) may lead to I/O bottlenecks and substantially lower performance.

For some problems 64-bit integers may be required. In Pascal it is `int64`. In C/C++ it is `long long` and its token for `scanf` /`printf` is `%lld`.

All tasks are divided into subtasks. You need to pass all test cases in a subtask to get points.

T201 - FONT SIZE

Time Limit: 1.000 s / Memory Limit: 256 MB

Winnie has finally finished her homework! It is an essay and contains N characters. To make her homework stand out from the others', she tries to put efforts into the font size of those characters and prettify the essay.

Winnie is an artistic person. She has come up with M constraints for the font size such that if all of those constraints are satisfied, the essay will become so stunning that it is definitely going to receive "SSS+" grade from her teacher.

The i^{th} constraints can be represented as a tuple (L_i, R_i, S_i):
If $S_i > 0$, it means that all characters between position L_i to R_i inclusive should have font size equal to S_i.
If $S_i = 0$, it means that there should be a pair of (p, q), such that $L_i \leq p < q \leq R_i$ and the font size at position p is not equal to font size at position q.

Also, the font size of each character should be an integer between 1 to T inclusive, as Winnie doesn't want the characters to be too big.

Since Winnie is lazy, she finds her best friend, you, to help her adjust the font size. Please help her to find a way to set the font size satisfying all M constraints or simply tell her it is impossible.

INPUT

The first line contains three integers, N, M, T, where N is the number of character in the essay, M is the number of constraints and T is the maximum font size.
Then, the i^{th} of the next M lines contain three integers L_i, R_i and S_i, which represent the i^{th} constraint explained above.

OUTPUT

If there is no way to set the font size while satisfying all the constraints, output `NO` in a single line.

Otherwise, output two lines. The first line of output should be the string `YES`.
Then output N integers in the second line, where the i^{th} integer denotes the font size of the character at position i in the essay. Please be reminded that the font sizes must be between 1 and T inclusive. If there are multiple solutions, output any one.

SAMPLE TESTS

	Input	Output
1	10 3 5 2 5 3 8 9 0 7 8 1	YES 4 3 3 3 3 5 1 1 2 4
2	3 4 2 1 1 1 3 3 2 1 2 0 2 3 0	NO

"1 1 2" does not satisfy constraint 3 and
"1 2 2" does not satisfy constraint 4

SUBTASKS

For all cases:
$1 \le N, M, T \le 10^5$
$1 \le L_i \le R_i \le N$ for $i = 1, 2, \ldots, N$
$0 \le S_i \le T$ for $i = 1, 2, \ldots, N$

	Points	Constraints
1	3	$S_i = 0$ $1 \le N, M \le 1000$
2	5	$S_i > 0$ $1 \le N, M \le 1000$
3	10	$1 \le N, M \le 1000$ $10 < T \le 10^5$
4	10	$10 < T \le 10^5$
5	25	$1 \le N, M \le 1000$
6	47	No additional constraints

T202 - LET'S LIFT STRING SWIFTLY

Time Limit: 1.000 s / Memory Limit: 256 MB

Recently, a new game named LLSS (Let's Lift String Swiftly) has gone viral in Hackerland. The description of the game is as follows:

> Given a string S and a string T, both of them consist of lowercase letter(s) only. Also, 2 integers C and D are given. You are going to convert string S to string T using the following operations:
>
> - Operation 1: Rearrange string S to an ordering of your choice. This operation can be used at most once, and can only be used before using Operations 2 and 3. This operation has no cost.
> - Operation 2: Insert a character of your choice into the **CURRENT** string S at a position of your choice. If the character is inserted after the x^{th} (1-based) character, the operation would incur a cost of $x \times C$. Notice that if you insert a character before the first character of string S, the cost is $0 \times C = 0$.
> - Operation 3: Delete a character of your choice from the **CURRENT** string S. If you delete the x^{th} (1-based) character, the operation would incur a cost of $x \times D$.
>
> You can apply Operations 2 and 3 one after another for any number of times and in any order. It is not necessary to use all 3 kinds of operations. Try your best to reduce your cost of conversion as much as you can!

Your friend, Clerith, loves LLSS very much. She would like to improve her skills in this game. However, she doesn't know how to achieve the minimum cost of the conversion. Therefore, she asks for your help.

She has prepared a string S and a string T. To train her skills, she has also prepared Q pairs of insertion cost and deletion cost. The i^{th} pair has an insertion cost $C = C_i$ and a deletion cost $D = D_i$. She would like you to calculate the minimum cost of converting string S to string T for all Q pairs so that she would know how much she can improve.

INPUT

The first line contains a string S.
The second line contains a string T.
The third line contains an integer Q.
For the next Q lines, the i^{th} line contains 2 integers, C_i and D_i, representing the insertion cost and the deletion cost of i^{th} pair.

OUTPUT

Output Q lines.
The i^{th} line should contain an integer that is the minimum cost of converting string S to string T with insertion cost C_i and deletion cost D_i.

SAMPLE TESTS

	Input	Output
1	bdce bac 1 2 6	14

For the first and only pair, the inserting cost is 2 and the deleting cost is 6.
One way to convert string S to string T is as follows:

1. Rearrange string S to be debc.
2. Delete the first character of string S, costing $1 \times 6 = 6$. String S becomes ebc.
3. Delete the first character of string S again, costing $1 \times 6 = 6$. String S becomes bc.
4. Insert a character a after the first character of string S, costing $1 \times 2 = 2$. String S becomes bac and has been converted to string T successfully.

The total cost is $6 + 6 + 2 = 14$, which is the minimum cost of converting string S to string T for the first pair of cost.

2	yougonnagivehardwork nevergonnagiveyouup 1 49758623 99517246	1940586297

SUBTASKS

For all cases:
Denote $|S|$ = length of string S and $|T|$ = length of string T.
$1 \le |S|, |T|, Q \le 3 \times 10^5$
$0 \le C_i, D_i \le 10^8$

	Points	Constraints
1	17	$Q = 1$ $1 \le \|S\|, \|T\| \le 16$
2	15	$C_i = D_i$ for all i.
3	25	$Q = 1$
4	43	No additional constraints

T203 - ENTROPY MAXIMIZER

Time Limit: 1.000 s / Memory Limit: 256 MB

Charlie is a cryptography expert devising a new cryptography algorithm. Typically, the quality of a cryptography algorithm can be measured by the amount of entropy (randomness) it introduces to the data to be encrypted.

Charlie's new algorithm uses many partial keys to form a master key for encryption and decryption.. The partial keys content can be represented by N strings $S_1, S_2, \ldots, S_N$, all of which contain `a` and `b` only. To make a master key, he needs to choose some (at least 1 but not necessarily all) of these N partial keys (strings), rearrange them arbitrarily, then concatenate them into a new master key T (also a string). For example, if there are partial keys `aaba`, `bba`, `ab` and `a`, one possible master key would be `bba`-`a`-`aaba`.

He defines the entropy of the master key (concatenated string) T as the maximum sum of length of strings chosen when he chooses at most K non-overlapping substrings of T that contains only the character `a`. Note that a substring of a string T is any string that can be obtained from T by removing some or none of its prefix and suffix. Using the example above, the master key has entropy = 4 if $K = 1$ and entropy = 4 + 1 = 5 when $K = 2$.

Help Charlie find a master key obtainable from the N partial keys S_i that has the maximum entropy.

INPUT

The first line consists of two integers N and K.

N lines follows. The i^{th} of which contains string S_i, consisting of only `a` and `b`.

OUTPUT

On the first line, output a single integer E, the maximum entropy that can be obtained.

On the second line, output the concatenated string T, separating the strings S_i used with `-` (ASCII 45). If there are multiple solutions that can achieve the same maximum entropy E, output any one.

SAMPLE TESTS

	Input	Output
1	2 1 aaaab abaa	6 abaa-aaaab

You can obtain `aa-aaaa`, which has length 6.

	Input	Output
2	1 1 bbb	0 bbb

You must use at least 1 S_i.

	Input	Output
3	5 2 bbaaa babaaaa aaaababb aaaaaabab bbbbb	17 bbaaa-aaaababb-babaaaa-aaaaaabab

2 different substrings, `aaa-aaaa` and `aaaa-aaaaaa` can be obtained.
Notice that you may not need to use all the N strings.

SUBTASKS

For all cases:
$1 \leq N \leq 10^5$
$1 \leq K \leq 2$
$1 \leq$ Sum of length of $S_i \leq 10^6$

	Points	Constraints
1	15	$1 \leq N \leq 50$ $K = 1$
2	9	$K = 1$
3	11	$1 \leq N \leq 9$ $K = 2$ $1 \leq \lvert S_i \rvert \leq 10$ for all $1 \leq i \leq N$
4	23	$1 \leq N \leq 50$ $K = 2$
5	42	$K = 2$

T204 - METROS OF RUNETERRA

Time Limit: 30.000 s / Memory Limit: 256 MB

The Runeterra is a universe where Champions and Minions live in. There are 11 regions in Runeterra, namely Bilgewater, Demacia, Feljord, Ionia, Ixtal, Mount Targon, Noxus, Piltover, Shadow Isles, Shurima and Zaun. As the population grows, Runeterra is clearly lacking public transport infrastructure. Minions in Runeterra relies on horse wagons and boats to travel. The government has decided to build a metro network within each region to better serve the minions.

The regions' geographies are very different so their metro networks should be designed independently. From now on let's consider a particular region.

There are N towns, numbered $1, 2, \ldots, N$, in the region where minions can live, study and work in. If overlay the map on a Cartesian coordinate plane, town i is located at (X_i, Y_i). No three towns are collinear. From the latest census results, we know that $P_{i,j}$ minions will travel between town i and town j on each day. Note that we do not need to consider their "home town" so $P_{i,j} = P_{j,i}$.

The government does not have a lot of gold. In fact, the budget only allows the government to build M train tracks. A track is a direct connection between two different towns. The length of the track would be the Euclidean distance between the two towns. Note that it is not allowed to build two tracks between the same pair of towns. Tracks can also cross one another. It is required to build exactly M tracks, but it is not necessary to connect all towns to the metro network. It is also allowed to have form multiple sub-networks that are not connected together.

Because Runeterra is full of dangers, minions will always travel using the shortest path on the metro network. They can only change trains at towns that are connected by two or more tracks. They cannot change trains at the intersections of the tracks. The efficiency of the network is measured by the satisfaction score. If town i and town j are connected by the metro network, then the satisfaction will increase by $1/d_{i,j}$ per minion where $d_{i,j}$ is the length of the shortest path between town i and j. If the total satisfaction exceeds the Target Satisfaction T, the minions will start carrying balloons with them!

You are tasked to design the metro networks. Garen has already designed the metro network for Demacia as an example for you. Now, you have 5 hours to design the metro networks for the other 10 regions.

INPUT

The first line contains 3 integers N, M and T.

The i^{th} of next N lines contains two integers X_i and Y_i.

Each of the next N lines contains N integers. The j^{th} integer on the i^{th} line is $P_{i,j}$.
$P_{i.i} = 0$ for $i = 1, 2, \ldots, N$. Also, $P_{i,j} = P_{j,i}$.

Please refer to the Test Case Overview section for the values of N, M and T.
For all cases, $-1000 \leq X_i, Y_i, \leq 1000$ and $0 \leq P_{i,j} \leq 10^6$. All (X_i, Y_i) are distinct and no three towns are collinear.

OUTPUT

Output the M tracks in separate lines to describe the metro network.

The i^{th} line should contain two integers A_i and B_i, indicating that a track should be built between town A_i and B_i.
You may output the tracks in any order, and the two towns of a track can be output in any order.

To read from file and write to file, you may either use file I/O or input redirection: `./programname < regionX.txt > metroX.txt`

Output-Only Task Information

This is an output-only task. Please place all the output files in the root of a zip file. Do not place them in a subfolder. Windows/*nix line endings will be corrected automatically. Maximum size of the zip file is 2^{24} bytes (16MB).

Output filename requirements: `metro1.txt` `metro2.txt` `metro3.txt` `metro4.txt` `metro5.txt` `metro6.txt` `metro7.txt` `metro8.txt` `metro9.txt` `metro10.txt`

SAMPLE TESTS

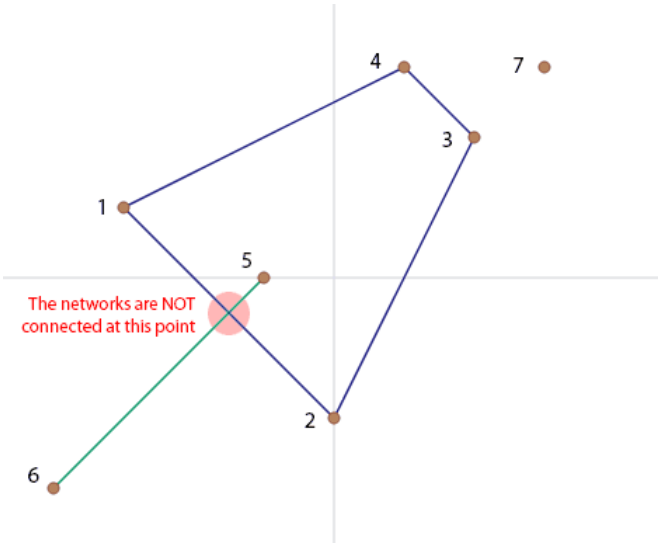
	Input	Output
1	7 5 100 -3 1 0 -2 2 2 1 3 -1 0 -4 -3 3 3 0 86 10 90 1 2 3 86 0 80 45 4 0 6 10 80 0 30 6 2 5 90 45 30 0 0 8 9 1 4 6 0 0 48 5 2 0 2 8 48 0 12 3 6 5 9 5 12 0	5 6 2 1 2 3 4 1 4 3

SCORING

Each test case carries 10 points. The total score of the task is the sum of the scores of the test cases.

For a test case, if T is the Target Satisfaction and S is the satisfaction that your metro network obtained, the score for the test case will be $\min(10, 10 \times (\frac{S}{T})^3)$. It is guaranteed that there exists a solution that achieves satisfaction greater than T, but that particular solution may not necessarily connect all N towns to the metro network.

EXPLANATION



The total satisfaction is 100.154. The score is $\min(10, 10 \times (100.154/100)^3) = \min(10, 10.005) = 10$

Min Distance Satisfaction	2	3	4	5	6	7
1	4.243 20.270	5.886 1.699	4.472 20.125	X	X	X
2		4.472 17.889	5.886 7.645	X	X	X
3			1.414 21.213	X	X	X
4				X	X	X
5					4.243 11.314	X
6						X

TEST CASE OVERVIEW

Case	Input	Output	N	M	T
1	region1.txt	metro1.txt	7	8	12000
2	region2.txt	metro2.txt	8	7	5000
3	region3.txt	metro3.txt	8	13	7500
4	region4.txt	metro4.txt	15	20	35000
5	region5.txt	metro5.txt	35	20	22500
6	region6.txt	metro6.txt	30	38	150000
7	region7.txt	metro7.txt	40	40	28000
8	region8.txt	metro8.txt	49	60	100000
9	region9.txt	metro9.txt	60	38	80000
10	region10.txt	metro10.txt	80	100	400000