Problem 12: Bruh I just realized we made another grid problem

10+5+2 Points

Problem ID: `multiplication`
Rank: `3+4+4`

Introduction

To learn multiplication, Big Ben printed a gigantic poster of a multiplication table created from his favorite integers collected over CALICO's 5 year history. Unfortunately the poster was damaged and now he's very sad :(Many entries were lost, and many others were corrupted. Big Ben now has only a vague memory of his favorite off diagonal cells.

Big Ben is working on a new multiplication table. He doesn't expect a perfect recreation of his old one, and he's happy tolerating a few different entries. Help Big Ben pick numbers for his new multiplication table!

Problem Statement

Your task is to construct a list of $\mathbf{N}$ integers $a_1, a_2, \ldots, a_{\mathbf{N}}$, which defines a $\mathbf{N} \times \mathbf{N}$ multiplication table where the entry at row i and column j is $a_i a_j$.

You are given $\mathbf{M}$ constraints, each consisting of:

- A row index $\mathbf{R}_i$ in the list $\mathbf{R}_1, \mathbf{R}_2, \ldots, \mathbf{R}_{\mathbf{M}}$ and a column index $\mathbf{C}_i$ in the list $\mathbf{C}_1, \mathbf{C}_2, \ldots, \mathbf{C}_{\mathbf{M}}$
  - Row and indices are guaranteed to be off-diagonal, $\mathbf{R}_i \neq \mathbf{C}_i$.
- A value $\mathbf{V}_i$ in the list $\mathbf{V}_1, \mathbf{V}_2, \ldots, \mathbf{V}_{\mathbf{M}}$
- A weight $\mathbf{W}_i$ in the list $\mathbf{W}_1, \mathbf{W}_2, \ldots, \mathbf{W}_{\mathbf{M}}$

Additionally, you may select a subset $S \subseteq \{1, 2, \ldots, \mathbf{M}\}$ of d constraints to discard, up to $\mathbf{D}$.

Your goal is to minimize the total $\mathbf{W}_i$-weighted relative error between the table entries $a_{\mathbf{R}_i} a_{\mathbf{C}_i}$ and the corresponding values $\mathbf{V}_i$, taken over all non-discarded constraints. In other words, minimize:

$$\text{Penalty} = \sum_{\substack{i=1 \\ i \notin S}}^{\mathbf{M}} \mathbf{W}_i \cdot \frac{|a_{\mathbf{R}_i} \cdot a_{\mathbf{C}_i} - \mathbf{V}_i|}{\mathbf{V}_i}$$

Important Note

The grading for this problem is unique!

Finding the exact optimal answer for this problem is quite hard! As such, your goal isn't to find the exact optimal answer, but to simply achieve a better penalty than the penalty of solutions created by different AI models.

Your submission will be accepted only if for all test cases it achieves a penalty no higher than the penalty of the solution created by the AI model for that test set.

There are additional longer samples available to download that aren't shown in the statement

Input Format

Each test file describes a single test case, which contains two lines:

- The first line of the input contains three space-separated integers $\mathbf{N}$ $\mathbf{M}$ $\mathbf{D}$ denoting the size of the table, the number of observed cells, and the number of observations we can discard, respectively.
- The next $\mathbf{M}$ lines each describe a constraint and each contain four space-separated integers $\mathbf{R_i}$ $\mathbf{C_i}$ $\mathbf{V_i}$ $\mathbf{W_i}$ denoting the row index, column index, value, and weight respectively.
    - The given constraints are 1-indexed: the first constraint contains $\mathbf{R_1}$, the second contains $\mathbf{R_2}$, and so on.

Note that multiple constraints may have the same $\mathbf{R_i} = \mathbf{R_j}$ and $\mathbf{C_i} = \mathbf{C_j}$. In these cases, both contribute to the penalty.

Output Format

For the single test case in each test file, output two lines:

- The first line should contain $\mathbf{N}$ space separated integers, $a_0\ a_1\ \ldots\ a_{\mathbf{N}}$
- The second line should contain $d + 1$ space separated integers denoting the number of discarded observations followed by the discarded observations themselves, $d\ S_1\ S_2\ \ldots\ S_d$

Constraints

Time Limit: **10 Seconds**

Input Constraints

$1 \leq \mathbf{N} \leq 4 \cdot 10^3$
$1 \leq \mathbf{M} \leq 2 \cdot 10^6$
$0 \leq \mathbf{D} \leq \mathbf{M}$

$1 \leq \mathbf{R_i}, \mathbf{C_i} \leq \mathbf{N}$
$\mathbf{R_i} \neq \mathbf{C_i}$
$1 \leq \mathbf{V_i} \leq 10^9$
$1 \leq \mathbf{W_i} \leq 10^3$

Note that multiple constraints may have the same $\mathbf{R_i} = \mathbf{R_j}$ and $\mathbf{C_i} = \mathbf{C_j}$. In these cases, both contribute to the penalty.

Output Constraints

$1 \leq a_i \leq 10^9$
$0 \leq d \leq \mathbf{D}$
$1 \leq S_i \leq \mathbf{M}$

Test Set 1

Your submission must achieve a penalty no worse than that achieved by the free online version of Grok 4.

Test Set 2

Your submission must achieve a penalty no worse than that achieved by GPT-5.4 without an agentic harness.

Test Set 3

Your submission must achieve a penalty no worse than that achieved by GPT-5.4 with a special agentic harness designed by a Codeforces Legendary Grandmaster.

Sample Test Cases

There are additional longer samples available to download that aren't shown in the statement

To download all sample test cases for this problem, find it in the problemset tab on CALICOJudge.

Sample Input

```
4 5 1
1 2 6 10
2 3 12 10
3 4 20 10
1 3 8 5
2 4 100 50
```

Sample Output

```
2 3 4 4
1 5
```

Note that this is one of many possible correct outputs. If there are multiple solutions, you may output any of them.

Sample Explanation

For this test case, a is `2 3 4 4` and a single constraint with index 5 is discarded.

- Constraint 1: $10 \cdot \frac{|(2 \cdot 3) - 6|}{6} = 0.00$
- Constraint 2: $10 \cdot \frac{|(3 \cdot 4) - 12|}{12} = 0.00$
- Constraint 3: $10 \cdot \frac{|(4 \cdot 4) - 20|}{20} = 10 \cdot \frac{4}{20} = 2.00$
- Constraint 4: $5 \cdot \frac{|(2 \cdot 4) - 8|}{8} = 0.00$
- Constraint 5: Discarded

Total Penalty: 2.00

For test set 1, suppose Grok 4 chose `1 1 1 1` and discarded 5.

- Penalty: $10(\frac{5}{6}) + 10(\frac{11}{12}) + 10(\frac{19}{20}) + 5(\frac{7}{8}) \approx 8.33 + 9.17 + 9.50 + 4.38 \approx$ **31.38**

For test set 2, suppose GPT-5.4 chose `8 10 1 10` and discarded 1.

- Penalty: $10(\frac{2}{12}) + 10(\frac{10}{20}) + 5(\frac{0}{8}) + 50(\frac{0}{100}) \approx 1.67 + 5.00 + 0.00 + 0.00 \approx$ **6.67**

For test set 3, suppose Agentic GPT-5.4 chose `4 7 2 14` and discarded 1.

- Penalty: $10(\frac{2}{12}) + 10(\frac{8}{20}) + 5(\frac{0}{8}) + 50(\frac{2}{100}) \approx 1.67 + 4.00 + 0.00 + 1.00 \approx$ **6.67**

Result: While your penalty (2.00) is not the absolute minimum possible for this array, it is strictly lower than the best baseline penalty (6.67). Since the grading only requires you to beat the AI, this output would successfully passes all three versions!