Big Queries ✓

Problem code: BGQRS

Tweet

Read problems statements in **Mandarin Chinese**, **Russian** and **Vietnamese** as well.

WARNING: This problem has large input / output files. Use of faster I/O methods is recommended.

You are given an array **A** consisting of **N** integers. You have to answer **M** queries on it. Each query belongs to one of the following three types:

- **1 L R X** : multiply each number in the range $A_L, A_{L+1}, ..., A_R$ by **X**.

- **2 L R Y** : Replace the elements $A_L, A_{L+1}, ..., A_R$ by **Y, 2 * Y, ... (R - L + 1) * Y**. In other words, the number A_i will be equal to **(i - L + 1) * Y** for each **i** from **L** to **R**.

- **3 L R** : Find the product of all numbers in the range $A_L, A_{L+1}, ..., A_R$. As this number could be very large, you have to just find the number of trailing zeros of this number when represented in decimal notation.

Input

The first line of the input contains an integer **T** denoting the number of test cases. **T** test cases follow.

The first line of each test case contains two space-separated integers **N** and **M**.

The second line contains **N** space-separated integers denoting $A_1, A_2, ..., A_N$

For next **M** lines, each line contains a query.

Each query is given by three or four (please refer to the statement) space separated integers.

The first integer denotes type of the query. For every type of query next two integers denote **L** and **R**. For each query of type 1, next integer denote **X**. For each query of type 2, next integer denote **Y**.

Output

For each test case, output a single line containing the sum of answers of all the queries of type 3.

Constraints

- $1 \leq T \leq 5$
- $1 \leq N, M, N + M \leq 10^5$
- $1 \leq L \leq R \leq N$
- $1 \leq X, Y, A_i \leq 10^9$

Subtasks

- Subtask #1 (18 points): $1 \leq N, M, X, Y, A_i \leq 10$
- Subtask #2 (31 points): $1 \leq N, M \leq 1000$
- Subtask #3 (51 points): **original constraints**

Example

Input:
```
1
5 5
2 4 3 5 5
3 2 4
3 2 5
2 2 4 1
1 3 3 10
3 1 5
```

Output:
```
5
```

Explanation

Array: **[2, 4, 3, 5, 5]**

- 1st query: **[4, 3, 5]**, 4 * 3 * 5 = 60 : answer 1.
- 2nd query: **[4, 3, 5, 5]**, 4 * 3 * 5 * 5 = 300 : answer 2.
- 3rd query: **[2, 4, 3, 5, 5] => [2, 1, 2, 3, 5]**.
- 4th query: **[2, 1, 2, 3, 5] => [2, 1, 20, 3, 5]**.
- 5th query: **[2, 1, 20, 3, 5]**, 2 * 1 * 20 * 3 * 5 = 600 - answer 2.

Sum of all answers = 5.