

## Problem 1. prob

Input file:           prob.in  
Output file:          prob.out  
Time limit:          1 second  
Memory limit:        256MB

有  $n$  道题目,  $m$  个有经验的队伍, 每个队伍可能做过一些题目.

现在, 请你判断, 我们是否可以选出一些题目, 使得在这些题目中, 对于每个队伍, 做过的题目小于等于没做过的题目.

### Input

第 1 行 1 个整数  $T$  表示数据组数;

对于每组数据:

第 1 行包含两个整数  $n$  和  $m$ , 分别表示题目数和队伍数;

接下来  $n$  行, 每行包含  $m$  个数字 (0 和 1), 表示该题是否被某个队伍做过 (第  $i$  个数字表示是否被第  $i$  个队伍做过).

### Output

对于每组数据, 输出"YES" 或者"NO", 表示是否能找到题目要求的题目集合.

### Sample

| prob.in | prob.out |
|---------|----------|
| 2       | NO       |
| 5 3     | YES      |
| 1 0 1   |          |
| 1 1 0   |          |
| 1 0 0   |          |
| 1 0 0   |          |
| 1 0 0   |          |
| 3 2     |          |
| 1 0     |          |
| 1 1     |          |
| 0 1     |          |

样例解释:

第一组数据: 因为第一个队伍做过所有题目, 所以条件不可能达成.

第二组数据: 可以选择第一道和第三道题目, 这样每个队伍做过的题目和没做的题目都是 1 道, 所以可以.

### Note

- 对于 30% 的数据,  $1 \leq m \leq 2$ ;
- 对于 100% 的数据,  $1 \leq m \leq 4$ ,  $1 \leq n \leq 10^5$ ,  $1 \leq T \leq 100$ , 所有组数据的  $n$  的和小于等于  $2 \times 10^5$ .

## Problem 2. pizza

Input file:            `pizza.in`  
Output file:         `pizza.out`  
Time limit:          1 second  
Memory limit:       256 MB

JJ 要买披萨.

有两种披萨, 且每种都有无限份, JJ 可以各买一些份数, 每份都要切成  $S$  片.

现在 JJ 家里来了  $N$  位客人, 第  $i$  位客人要吃  $s_i$  片披萨才能吃饱, 吃一片第一种披萨会有  $a_i$  的快乐值, 吃一片第二种披萨会有  $b_i$  的快乐值.

现在 JJ 要定披萨, JJ 不希望浪费, 所以 JJ 在让每位客人都能吃饱的前提下, 会买尽量少的披萨.

请问所有客人的快乐值之和最大能多少?

### Input

第 1 行 2 个整数  $N, S$  分别表示客人的数量和一份披萨能分成多少片.

接下来  $N$  行, 第  $i$  行三个数  $s_i, a_i, b_i$ .

### Output

输出一个数, 表示答案.

### Sample

| <code>pizza.in</code>           | <code>pizza.out</code> |
|---------------------------------|------------------------|
| 3 12<br>3 5 7<br>4 6 7<br>5 9 5 | 84                     |

样例解释: 只需要买一个披萨就够了, 如果买第一种披萨, 总的快乐值是  $5 \times 3 + 6 \times 4 + 9 \times 5 = 84$ , 买第二种披萨对应的快乐值是  $7 \times 3 + 7 \times 5 + 5 \times 5 = 81$ , 所以只需要买一个第一种类的披萨, 可以获得 84 的快乐值.

| <code>pizza.in</code>  | <code>pizza.out</code> |
|--|------------------------|
| 6 10<br>7 4 7<br>5 8 8<br>12 5 8<br>6 11 6<br>3 3 7<br>5 9 6 | 314                    |

### Note

- 对于 30% 的数据,  $1 \leq N \leq 1000, 1 \leq s_i \leq 10$ ;
- 对于 100% 的数据,  $1 \leq N, S, s_i, a_i, b_i \leq 10^5$ .

## Problem 3. scream

Input file:            scream.in  
Output file:           scream.out  
Time limit:           4 second  
Memory limit:        256 MB

你面前按顺序排列着  $n$  块冰淇淋 (种类可能不同), 你手中有  $m$  个冰淇淋桶, 你现在需要将这  $n$  块冰淇淋分成  $m$  份, 要求每份冰淇淋是连续的一段, 然后将这  $m$  份冰淇淋分别装入  $m$  个冰淇淋桶中.

第  $i$  个冰淇淋桶的美味值是装入其中的冰淇淋的种类数, 请问怎样分冰淇淋才能让美味值之和尽量大?

### Input

第 1 行包含 2 个整数  $n, m$ ;

第 2 行包含  $n$  个整数数  $a_1, a_2, \dots, a_n$  表示冰淇淋的种类.

### Output

输出一个整数表示美味值之和.

### Sample

| scream.in      | scream.out |
|----------------|------------|
| 4 1<br>1 2 2 1 | 2          |

样例说明: 只能将所有的冰淇淋装入一个桶中, 因为有两种冰淇淋, 所以美味值位 2.

| scream.in            | scream.out |
|----------------------|------------|
| 7 2<br>1 3 3 1 4 4 4 | 5          |

样例说明: 最优分法是  $\{1, 3\}, \{3, 1, 4, 4, 4\}$ , 美味值分别为 2, 3, 和为 5.

| scream.in              | scream.out |
|------------------------|------------|
| 8 3<br>7 7 8 7 7 8 1 7 | 6          |

### Note

- 对于 30% 的数据,  $1 \leq n \leq 1000$ ;
- 对于 100% 的数据,  $1 \leq n \leq 20000, 1 \leq a_i \leq n, 1 \leq m \leq \min(50, n)$ .