Help R2-D2!

In Episode III of Star Wars (whose alleged title is "How I became Vader"), R2-D2 (Artoo-Detoo) is again confronted to a tedious work. He is responsible for the loading of the republic transport starships in the fastest way. Imagine a huge space area where n starships are parked. Each starship has a capacity of K cubic femtoparsec. Containers C_i arrive one at a time with some volume v_i (expressed in cubic femtoparsec). R2-D2 wants to minimize the number of starships used for a given sequence of containers.

Smart as he is, R2-D2 knows for sure that the problem is a hard one, even with the force being around. Here is the heuristics he selected to solve his problem. Start with all starships ready to load, and numbered S_0,S_1,etc. When a container C_j arrives, select the starship of minimal index i that can contain C_j and put it in S_i. In some sense, this heuristic minimizes the move of the container arriving before its loading.

At the end of the n arrivals, R2-D2 counts the number s of starships used and he measures the total waste w of the sequence. For i=0..s-1, the waste in starship i is given by the unused volume.

Your task is to simulate the algorithm of R2-D2.

Input

The first line of the input contains a number T ≤ 10 that indicates the number of test cases to follow. Each test case begins with K on a line (K ≤ 1000), followed by the number of containers in the sequence, n, on the second line (1 ≤ n ≤ 1000000). There are two possible formats for the remaining lines. If it contains one integer, then this is the next v_i. If it begins with the character b (for block), it is followed by 2 integers r and v. This means that the r next containers arriving have volume v.

Output

Your program must output the number s of starships used, followed by a blank, followed by the total waste w.

You can assume, that at most 100000 starships are needed, and R2-D2 has to change the starships in which the next container is loaded at most 100000 times.

Example

Input:

```
2
100
3
50
25
70
100
4
50
b 2 40
```

```
20
```

Output:

```
2 55
2 50
```