Elections

Your goal is to write a program that will count the number of seats won by each party in the elections. We have n parties and s seats, the parties are numbered with consecutive numbers in $[n]$ (starting from 1). The seats are distributed using the *d'Hondt method*. The method works as follows. After all the votes have been counted, successive quotients are calculated for each party. The party with the largest quotient wins one seat, and its quotient is recalculated. This is repeated until the required number of seats is filled. The formula for the quotient is

$$quot = \frac{votes}{seats + 1}$$

where:

- *votes* is the total number of votes that party received,
- *seats* is the number of seats that party has been allocated so far, initially 0 for all parties.

In the case of a tie, the seat goes to the party with the smallest number.

You should implement an algorithm that works in $O((s + n) \log n)$-time.

Input

The first line contains integer z ($1 \leqslant z \leqslant 2 \cdot 10^9$) – the number of data sets. Each data set is as follows:

The first line contains number n – the number of parties and number s – the number of seats ($1 \leqslant n, s \leqslant 4000000$). The next line contains n integers with the number of votes received by subsequent parties, separated by a space.

Output

The total number of seats obtained by each party, separated by a space.

Example

For the input:

```
3
5 7
20 10 100 20 30
5 6
20 10 100 20 30
5 8
20 10 100 20 30
```

the output is:

```
1 0 5 0 1
1 0 4 0 1
1 0 5 1 1
```