H • Powers of Pascal

The *Pascal matrix* is the (infinite) matrix defined by (zero based row and column):

`Pascal[row, column] = Comb(row, column) for 0 <= column <= row`

and zero otherwise, where `Comb(n, k)` is the number of combinations of `n` things taken `k` at a time (the binomial coefficient).

```
1   0    0    0    0    0    0    0    0    0 ...
1   1    0    0    0    0    0    0    0    0 ...
1   2    1    0    0    0    0    0    0    0 ...
1   3    3    1    0    0    0    0    0    0 ...
1   4    6    4    1    0    0    0    0    0 ...
1   5   10   10    5    1    0    0    0    0 ...
1   6   15   20   15    6    1    0    0    0 ...
1   7   21   35   35   21    7    1    0    0 ...
1   8   28   56   70   56   28    8    1    0 ...
1   9   36   84  126  126   84   36    9    1 ...
 .   .    .    .    .    .    .    .    .    .
 .   .    .    .    .    .    .    .    .    .
 .   .    .    .    .    .    .    .    .    .
```

For this problem, you will write a program to compute entries in powers of the *Pascal* matrix:

$$\text{Pascal}^P = \text{Pascal} \times \text{Pascal} \times \ldots \times \text{Pascal} \text{ (P factors)}$$

Since the matrix is lower triangular, all powers are lower triangular and only the upper left `N` by `N` corner is used in computing coefficients in the upper left `N` by `N` corner of the power.

Input

The first line of input contains a single integer K, ($1 \le K \le 1000$), which is the number of data sets that follow. Each data set should be processed identically and independently.

Each data set consists of a single line of input containing four space-separated decimal integers. The first integer is the data set number. The second integer is the power, `P` (`1 <= P <= 100,000`), to which to raise the Pascal matrix. The third and fourth integers give the row number, `R`, and the column number, `C`, of the desired entry (`0 <= C <= R <= 100,000`).

Output

For each data set there is a single line of output. The line consists of the data set number, a single space, which is then followed by the requested entry of the requested *Powers of the Pascal* matrix. Input values will be restricted so results will not overflow a 64-bit integer value.

Sample Input	Sample Output
3	1 56
1 1 8 3	2 8759577256290
2 9 21 13	3 199998000000000
3 200 100000 99998	