F. Let's Play the Hat?

time limit per test

2 seconds

memory limit per test

256 megabytes

input

standard input

output

standard output

*The Hat is a game of speedy explanation/guessing words (similar to Alias). It's fun. Try it! In this problem, we are talking about a variant of the game when the players are sitting at the table and everyone plays individually (i.e. not teams, but individual gamers play).*

n people gathered in a room with m tables (n≥2m). They want to play the Hat k times. Thus, k games will be played at each table. Each player will play in k games.

To do this, they are distributed among the tables for each game. During each game, one player plays at exactly one table. A player can play at different tables.

Players want to have the most "fair" schedule of games. For this reason, they are looking for a schedule (table distribution for each game) such that:

* At any table in each game there are either ⌊nm⌋ people or ⌈nm⌉ people (that is, either n/m rounded down, or n/m rounded up). Different numbers of people can play different games at the same table.
* Let's calculate for each player the value bi — the number of times the i-th player played at a table with ⌈nm⌉ persons (n/m rounded up). Any two values of bi must differ by no more than 1. In other words, for any two players i and j, it must be true |bi−bj|≤1.

For example, if n=5, m=2 and k=2, then at the request of the first item either two players or three players should play at each table. Consider the following schedules:

* First game: 1,2,3 are played at the first table, and 4,5 at the second one. The second game: at the first table they play 5,1, and at the second  —  2,3,4. This schedule is **not "fair"** since b2=2 (the second player played twice at a big table) and b5=0 (the fifth player did not play at a big table).
* First game: 1,2,3 are played at the first table, and 4,5 at the second one. The second game: at the first table they play 4,5,2, and at the second one  — 1,3. This schedule is **"fair"**: b=[1,2,1,1,1] (any two values of bi differ by no more than 1).

Find any "fair" game schedule for n people if they play on the mm tables of k games.

**Input**

The first line of the input contains an integer t (1≤t≤104) — the number of test cases in the test.

Each test case consists of one line that contains three integers nn, mm and kk (2≤n≤2⋅105,1≤m≤⌊n2⌋, 1≤k≤105) — the number of people, tables and games, respectively.

It is guaranteed that the sum of nk (n multiplied by k) over all test cases does not exceed 2⋅105.

**Output**

For each test case print a required schedule  — a sequence of kk blocks of m lines. Each block corresponds to one game, a line in a block corresponds to one table. In each line print the number of players at the table and the indices of the players (numbers from 1 to n) who should play at this table.

If there are several required schedules, then output any of them. We can show that a valid solution always exists.

You can output additional blank lines to separate responses to different sets of inputs.

**Example**

**input**

**Copy**

3

5 2 2

8 3 1

2 1 3

**output**

**Copy**

3 1 2 3

2 4 5

3 4 5 2

2 1 3

2 6 2

3 3 5 1

3 4 7 8

2 2 1

2 2 1

2 2 1