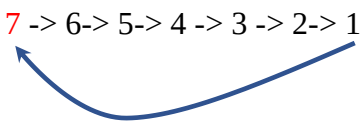
Problem: Who will get one year of free dedicated parking spot!

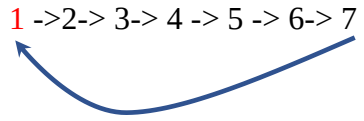
Monster campus decided to award a dedicated parking spot to a student for free for the whole year to their choice of garage. To determine the winner, the campus is hosting a quirky game. Since parking spots on campus are scarce and expensive, students are eager to compete for the chance to win.

There are exactly 10 garages on campus, numbered from 1 to 10. Participating students must choose one garage, and they receive a sequence number based on the order in which they register for their preferred garage. However, some garages may be less desirable due to their locations, so not all of them may be selected by the students. Ultimately, the students are grouped into G groups *(where* $G \leq 10$*)*, based on their unique garage choices, and these groups are assembled on a large campus field. Each group is identified by the garage number they selected, though some garages may be left unchosen.

Game Day! Each group g_i consists of n_i $(n_i \geq 2)$ students. Every student in the group is assigned a sequence number from 1 to n_i. These students form a circular line, waiting to be eliminated according to the game's rules. However, due to a distraction, it was discovered that the students in each group were standing in reverse order instead of the correct sequence. For example, if a group had 7 students, their positions were reversed, as shown in the figure below. (let us say number of students in this group is 7):



After realizing the wrong order of sequence, they reversed the circle to the correct order (note that they have not just changed their sequence number, they have physically changed their order) :



The elimination process is divided into two phases, explained below.

Phase1 Elimination:

In Phase1, eliminations occur within each group. For a given group g_i, the elimination phase continuous until the number of students in the group is reduced to a threshold th_i $(th_i < n_i)$.

The elimination process begins with the first student in the group's circle and moves in a fixed direction around the circle. In each step, a certain number of students k_i *-1* ($k_i > 0$) are skipped and the next student is eliminated. This process continues, with the circle shrinking as students

are progressively removed. As mentioned, the elimination process for the group g_i will stop when the group has th_i number of soldiers.

In summary, for a group g_i, you have the total number of students n_i ($n_i \geq 2$), and a number k_i ($k_i > 0$), which indicates that k_i-1 students are skipped and k_ith student is eliminated in circle. There is a threshold th_i ($th_i < n_i$), that indicates to stop eliminating in phase 1 when the number of students of the group reduced to th_i. The same process should be applied to all the groups according to their own n, k and th. The students soldiers will be transferred to phase 2.

Phase 1 Example

For example, if a group has n = 5, k = 2, and th = 2 then the phase 1 process will be the following:
First, the student at position 2 is eliminated, then the student at position 4 is eliminated, then student at position 1 is eliminated. As there are only two students remaining (students 3 and 5), the elimination process for this phase for the group will stop. The remaining students will be transferred to phase 2

Another example: if a group has n = 7, k = 3, and th = 3, then the phase 1 process will be the following:

The students at positions 3, 6, 2, and 7 will be eliminated and the remaining students 1, 4, and 5 will be transferred to phase 2.

Phase2 Elimination:

In Phase 2, eliminations occur across all groups using the following strategy:

- The student with the highest sequence number standing at the front of any group will be eliminated.
- If multiple students share the same highest sequence number at the front of different groups, the tie is broken by eliminating the student from the group with the smallest garage number (group number).
- This process continues until only one student remains across all the groups, and that student will be declared the winner!

Your task is to determine which student will win. Refer to the sample input/output for further clarification.

Input Specification (input must be taken using standard input using scanf.):

Note that if you use file i/o in any version of your submission, there will be -100 penalty.

The first line of the input contains the number of groups G(it means students have picked G unique garages from the 10 garages in the campus). The next G lines contain the information of the groups.