CodeGolf_v2022.1.0

0000, January 16, 2022 to 0000 February 16, 2022

(1-month-long Contest | 5 Problems)

1. **Rules:**
    a. This is an Individual Contest. No team submissions are allowed. Answer as many problems as you can.
    b. Solutions that compile without errors, pass all tests, and have the lowest character count win.
    c. For correct solutions, with equal character counts, the one submitted earlier gets a higher score/rank.
    d. Any language, supported by CodeChef, can be used. A full list of supported languages can be found here.
    e. Common Loopholes are forbidden, by default.
    f. Use of built-in libraries is allowed, provided CodeChef supports the libraries and that, they do not violate Rule e, above.
    g. No penalties for incorrect solutions.
    h. Multiple submissions are allowed.

2. **Problems:**

A. Palindrome

For a given positive integer K of not more than 1000000 digits, write the value of the smallest palindrome larger than K to output. Numbers are always displayed without leading zeros.

a. Input: The first line contains integer t, the number of test cases. Followed by t lines containing integers K.
b. Output: For each K, output the smallest palindrome larger than K.

B. Entrance Exam

The faculty of application management and consulting services (FAMCS) of the Berland State University (BSU) has always been popular among Berland's enrollees. This year, **N** students attended the entrance exams, but no more than **K** will enter the university. In order to decide

who are these students, there are series of entrance exams. All the students with score strictly greater than at least **(N-K)** students' total score gets enrolled.

In total there are **E** entrance exams, in each of them one can score between **0** and **M** points, inclusively. The first **E-1** exams had already been conducted, and now it's time for the last tribulation.

Sergey is the student who wants very hard to enter the university, so he had collected the information about the first **E-1** from all **N-1** enrollees (i.e., everyone except him). Of course, he knows his own scores as well.

In order to estimate his chances to enter the University after the last exam, Sergey went to a fortune teller. From the visit, he learnt about scores that everyone except him will get at the last exam. Now he wants to calculate the minimum score he needs to score in order to enter to the university. But now he's still very busy with minimizing the amount of change he gets in the shops, so he asks you to help him.

Input

The first line of the input contains an integer **T** denoting the number of test cases. The description of **T** test cases follows.

The first line of each test case contains four space separated integers **N, K, E, M** denoting the number of students, the maximal number of students who'll get enrolled, the total number of entrance exams and maximal number of points for a single exam, respectively.

The following **N-1** lines will contain **E** integers each, where the first **E-1** integers correspond to the scores of the exams conducted. The last integer corresponds to the score at the last exam, that was predicted by the fortune-teller.

The last line contains **E-1** integers denoting Sergey's score for the first **E-1** exams.

Output

For each test case, output a single line containing the minimum score Sergey should get in the last exam in order to be enrolled. If Sergey doesn't have a chance to be enrolled, output "Impossible" (without quotes).

Constraints

- $\mathbf{1 \le T \le 5}$
- $\mathbf{1 \le K < N \le 10^4}$
- $\mathbf{1 \le M \le 10^9}$
- $\mathbf{1 \le E \le 4}$

C. Appy and Contest

Appy and Chef are participating in a contest. There are N problems in this contest; each problem has a unique problem code between 1 and N inclusive. Appy and Chef decided to split the problems to solve between them — Appy should solve the problems whose problem codes are divisible by A but not divisible by B , and Chef should solve the problems whose problem codes are divisible by B but not divisible by A (they decided to not solve the problems whose codes are divisible by both A and B).

To win, it is necessary to solve at least K problems. You have to tell Appy whether they are going to win or lose.

Input

- The first line of the input contains a single integer T denoting the number of test cases. The description of T test cases follows.
- The first and only line of each test case contains four space-separated integers N, A, B, and K.

.Output

For each test case, print a single line containing the string "Win" if they can solve at least K problems or "Lose" otherwise (without quotes).

Constraints

- $1 \le T \le 15$
- $1 \le K \le N \le 10^{18}$
- $1 \le A, B \le 10^{9}$

D. Chef and Easy Queries

Chef published a blog post and is now receiving many queries about it. On the day i, he receives Q_i queries. But Chef can answer at most k queries in a single day.

The chef always answers the maximum number of questions that he can on any given day (note however that this cannot be more than k). The remaining questions (if any) will be carried over to the next day.

Fortunately, after n days, the queries have stopped. Chef would like to know the first day during which he has some free time, i.e. the first day when he answered less than k questions.

Input:

- First line will contain T, the number of testcases. Then the testcases follow.
- The first line of each testcase contains two space separated integers n and k.
- The second line of each testcase contains n space separated integers, namely Q_1, Q_2, … Q_n.

Output:

For each testcase, output in a single line the first day during which chef answers less than

k questions.

Constraints

- $1 \leq T \leq 10^5$
- $1 \leq$ sum of n over all testcases $\leq 10^5$
- $1 \leq k \leq 10^8$
- $0 \leq Q_i \leq 10$

E. K Fibonnaci

Chef recently had been studying about Fibonacci numbers and wrote a code to print out the kth term of the Fibonacci series(1, 1, 2, 3, 5, 8, 13….). He was wondering whether he could write a program to generate the **kth** term for similar series. More specifically : **T(n, k) is 1 if n <= k and T(n, k) = T(n-1, k) + T(n-2, k) + T(n-3, k) … + T(n-k, k) if n > k** . Given n and k output **T(n, k)%(1000000007)** as the answer could be very large.

Input

Two integers, **N** and **K**

Output

One integer, the nth term of the series mod 1000000007

Constraints

$1 \leq N, K \leq 2*10^5$