

Problem 1: The Best Gift

(Taken from: Codeforces 609B)

Emily's birthday is next week and Jack has decided to buy a present for her. He knows she loves books so he goes to the local bookshop, where there are n books on sale from one of m genres.

In the bookshop, Jack decides to buy *two books of different genres*.

Based on the genre of books on sale in the shop, find the number of options available to Jack for choosing two books of different genres for Emily. Options are considered different if they differ in at least one book.

The books are given by the indices of the genres. The genres are numbered from 1 to m .

Input

The first line contains two positive integers n and m ($2 \leq n \leq 2 \cdot 10^5, 2 \leq m \leq 10$) - the number of books in the bookstore and the number of genres.

The second line contains a sequence a_1, a_2, \dots, a_n , where a_i ($1 \leq a_i \leq m$) equals the genre of the i -th book. It is guaranteed that there is at least one book of each genre.

Output

Print a single integer, the number of ways in which Jack can choose books. It is guaranteed that the answer does not exceed $2 \cdot 10^9$.

Examples

- **Input**

4 3
2 1 3 1

- **Output**

5

- **Input**

7 4
4 2 3 1 2 4 3

- **Output**

18

Problem 2: Popes

(Taken from: UVa 957)

On the occasion of Pope John Paul II's death, the American weekly magazine Time observed the largest number of Popes to be selected in a 100-year period was 28, from 867 (Adrian II) to 965 (John XIII). This is a very interesting piece of trivia, but it would be much better to have a program to compute that number for a period of any length, not necessarily 100 years. Furthermore, the Catholic Church being an eternal institution, as far as we can predict, we want to make sure that our program will remain valid per omnia secula seculorum.

Write a program that, given the list of years in which each Pope was elected and a positive number Y , computes the largest number of Popes that were in office in a Y -year period, and the year of election for the first and last Popes in that period. Note that, given a year N , the Y -year period that starts in year N is the time interval from the first day of year N to the last day of year $N + Y - 1$. In case of a tie, i.e., if there is more than one Y -year period with the same largest number of Popes, your program should only report the most ancient one.

Input

The first line of the input contains a positive integer Y , the number of years of the period we are interested in. The second line contains another positive integer, the number of Popes, P . Each of the remaining P lines contains the year of election of a Pope, in chronological order. We know that $P \leq 10^5$ and also that the last year L in the file is such that $L \leq 10^6$, and that $Y \leq L$.

Output

For each test case, write to the output a single line with three integers: the largest number of Popes in a Y -year period, the year of election of the first Pope in that period, and the year of election of the last Pope in the period.

Problem 3: Copying Books

(Taken from: UVa 714)

Before the invention of book-printing, it was very hard to make a copy of a book. All of the contents had to be re-written by hand by *scribers*. The scribe had been given a book and after several months he finished its copy. One of the most famous scribes lived in the 15th century and his name was Xaverius Endricas Remius Ontius Xendrianus (Xerox). Anyway, the work was very annoying and boring. And the only way to speed it up was to hire more scribes.

Once upon a time, there was a theater ensemble that wanted to play famous Antique Tragedies. The scripts of these plays were divided into many books and actors needed more copies of them, of course. So they hired many scribes to make copies of these books. Imagine you have m books (numbered $1, 2, \dots, m$) that may have different numbers of pages (p_1, p_2, \dots, p_m) and you want to make one copy of each of them.

Your task is to divide these books among k scribes, $k \leq m$. Each book can be assigned to a single scribe only, and every scribe must get a continuous sequence of books. That means there exists an increasing succession of numbers $0 = b_0 < b_1 < b_2 < \dots < b_{k-1} < b_k = m$ such that the i -th scribe gets a sequence of books with numbers between $b_{i-1} + 1$ and b_i . The time needed to make a copy of all the books is determined by the scribe who was assigned the most work. Therefore, our goal is to minimize the maximum number of pages assigned to a single scribe. Your task is to find the optimal assignment.

Input

The input consists of N cases. The first line of the input contains only positive integer N . Then follow the cases. Each case consists of exactly two lines. At the first line, there are two integers m and k , $1 \leq k \leq m \leq 500$. At the second line, there are integers p_1, p_2, \dots, p_m separated by spaces. All these values are positive and less than 10^7 .

Output

For each case, print exactly one line. The line must contain the input succession p_1, p_2, \dots, p_m divided into exactly k parts such that the maximum sum of a single part should be as small as possible. Use the slash character ('/') to separate the parts. There must be exactly one space character between any two successive numbers and between the number and the slash.

If there is more than one solution, print the one that minimizes the work assigned to the first scribe, then to the second scribe, etc. But each scribe must be assigned at least one book.