Mục lục

Bài 1. Pie

File dữ liệu vào:	`stdin`
File kết quả:	`stdout`
Hạn chế thời gian:	1 s
Hạn chế bộ nhớ:	256 MB

My birthday is coming up and traditionally I'm serving pie. Not just one pie, no, I have a number N of them, of various tastes and of various sizes. F of my friends are coming to my party and each of them gets a piece of pie. This should be one piece of one pie, not several small pieces since that looks messy. This piece can be one whole pie though. My friends are very annoying and if one of them gets a bigger piece than the others, they start complaining. Therefore all of them should get equally sized (but not necessarily equally shaped) pieces, even if this leads to some pie getting spoiled (which is better than spoiling the party). Of course, I want a piece of pie for myself too, and that piece should also be of the same size. What is the largest possible piece size all of us can get? All the pies are cylindrical in shape and they all have the same height 1, but the radii of the pies can be different.

Dữ liệu vào

One line with a positive integer: the number of test cases. Then for each test case:

- One line with two integers N and F with $1 \leq N, F \leq 10000$: the number of pies and the number of friends.
- One line with N integers r_i with $1 \leq r_i \leq 10000$: the radii of the pies.

Kết quả

For each test case, output one line with the largest possible volume V such that me and my friends can all get a pie piece of size V. The answer should be given as a floating point number rounding to 6 digits after the floating point.

Ví dụ

stdin	stdout

3
3 3
4 3 3
1 24
5
10 5
1 4 2 3 4 5 6 5 4 2
	25.132741
	3.141593
	50.265482

Bài 2. Aggressive cows

File dữ liệu vào:	`Standard Input`
File kết quả:	`Standard Output`
Hạn chế thời gian:	1s
Hạn chế bộ nhớ:	256 MB

Farmer John has built a new long barn, with N $(2 \leq N \leq 100,000)$ stalls. The stalls are located along a straight line at positions $x_1, \ldots, x_N$ $(0 \leq x_i \leq 1,000,000,000)$.

His C $(2 \leq C \leq N)$ cows don't like this barn layout and become aggressive towards each other once put into a stall. To prevent the cows from hurting each other, FJ wants to assign the cows to the stalls, such that the minimum distance between any two of them is as large as possible. What is the largest minimum distance?

Dữ liệu vào

t – the number of test cases, then t test cases follows.

- Line 1: Two space-separated integers: N and C
- Lines $2 \ldots N+1$: Line $i+1$ contains an integer stall location, x_i

Kết quả

For each test case output one integer: the largest minimum distance.

Ví dụ

Standard Input	Standard Output

1
5 3
1
2
8
4
9
	3

Giải thích

FJ can put his 3 cows in the stalls at positions 1, 4 and 8, resulting in a minimum distance of 3.

Bài 3. Copying Books

File dữ liệu vào:	`Standard Input`
File kết quả:	`Standard Output`
Hạn chế thời gian:	1s
Hạn chế bộ nhớ:	256 MB

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

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 scribers to make copies of these books. Imagine you have m books (numbered $1, 2, \ldots, m$) that may have different number of pages ($p_1, p_2, \ldots, p_m$) 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 scriber only, and every scriber must get a continuous sequence of books. That means, there exists an increasing succession of numbers $0 = b_0 < b_1 < b_2, \ldots < b_{k-1} \leq b_k = m$ such that i-th scriber 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 scriber who was assigned the most work. Therefore, our goal is to minimize the maximum number of pages assigned to a single scriber. Your task is to find the optimal assignment.

Dữ liệu vào

The input consists of N cases (equal to about 200). 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 $p_1, p_2, \ldots, p_m$ separated by spaces. All these values are positive and less than 10000000.

Kết quả

For each case, print exactly one line. The line must contain the input succession $p_1, p_2, \ldots, p_m$ 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 scriber, then to the second scriber etc. But each scriber must be assigned at least one book.

Ví dụ

Standard Input	Standard Output

2
9 3
100 200 300 400 500 600 700 800 900
5 4
100 100 100 100 100
	100 200 300 400 500 / 600 700 / 800
	900
	100 / 100 / 100 / 100 100

Bài 4. Eko

File dữ liệu vào:	`Standard Input`
File kết quả:	`Standard Output`
Hạn chế thời gian:	1s
Hạn chế bộ nhớ:	256 MB

Lumberjack Mirko needs to chop down M metres of wood. It is an easy job for him since he has a nifty new woodcutting machine that can take down forests like wildfire. However, Mirko is only allowed to cut a single row of trees.

Mirko's machine works as follows: Mirko sets a height parameter H (in metres), and the machine raises a giant sawblade to that height and cuts off all tree parts higher than H (of course, trees not higher than H meters remain intact). Mirko then takes the parts that were cut off. For example, if the tree row contains trees with heights of 20, 15, 10, and 17 metres, and Mirko raises his sawblade to 15 metres, the remaining tree heights after cutting will be 15, 15, 10, and 15 metres, respectively, while Mirko will take 5 metres off the first tree and 2 metres off the fourth tree (7 metres of wood in total).

Mirko is ecologically minded, so he doesn't want to cut off more wood than necessary. That's why he wants to set his sawblade as high as possible. Help Mirko find the maximum integer height of the sawblade that still allows him to cut off at least M metres of wood.

Dữ liệu vào

The first line of input contains two space-separated positive integers, N (the number of trees, $1 \leq N \leq 1000000$) and M (Mirko's required wood amount, $1 \leq M \leq 2000000000$).

The second line of input contains N space-separated positive integers less than 1 000 000 000, the heights of each tree (in metres). The sum of all heights will exceed M, thus Mirko will always be able to obtain the required amount of wood.

Kết quả

The first and only line of output must contain the required height setting.

Ví dụ

Standard Input	Standard Output

`4 7`
`20 15 10 17`
	`15`

`5 20`
`4 42 40 26 46`
	`36`

Bài 5. Fibonacci Words

File dữ liệu vào:	`stdin`
File kết quả:	`stdout`
Hạn chế thời gian:	1 s
Hạn chế bộ nhớ:	256 MB

The Fibonacci word sequence of bit strings is defined as:

$$F(n) = \begin{cases} 0 & \text{if } n = 0 \\ 1 & \text{if } n = 1 \\ F(n-1) + F(n-2) & \text{if } n \geq 2 \end{cases}$$

Here denotes concatenation of strings. The first few elements are:

n	F(n)
0	0
1	1
2	10
3	101
4	10110
5	10110101
6	1011010110110
7	101101011011010110101
8	1011010110110101101011011010110110
9	1011010110110101101011011010110110101101011011010110101

Given a bit pattern p and a number n, how often does p occur in $F(n)$?

Dữ liệu vào

The first line of each test case contains the integer n ($0 \leq n \leq 100$). The second line contains the bit pattern p. The pattern p is nonempty and has a length of at most 100 000 characters.

Kết quả

For each test case, display its case number followed by the number of occurrences of the bit pattern p in $F(n)$. Occurrences may overlap. The number of occurrences will be less than 2^{63}.

Ví dụ

stdin	stdout
6 10	Case 1: 5