

Problem H. Stock Exchange

Input file: stock.in
Output file: standard output
Balloon Color: Blue

You have just started investing in a stock, and an insider told you that there's a list of N prices (in dollars) which will be assigned to that stock in the next N days (in some order which no one knows yet). Each price in that list will be assigned to exactly one of the next N days, and the price within the same day won't change. Your insider friend accepted to do any assignments you want, so you want to assign the prices to the days to maximize your profit.

Let's say you have already assigned the N prices to the N days. Before day 1, you already own 1 unit of that stock and you have no money at all. In each of the following N days, you can either sell (might be partial units) some of the stocks you own (and you get money in dollars), or you can buy (might be partial units) some of the available stocks (and you pay in dollars), using the price assigned to that day. You can also do nothing in any given day. Assume that there's an infinite supply of stocks in each day.

Your profit at the end is the amount of money you have, so you need to sell everything by the end of the N days.

Given the list of prices, your task is to arrange them somehow as described above to get the maximum profit.

Input

Your program will be tested on one or more test cases. The first line of the input will be a single integer T ($1 \leq T \leq 100$) representing the number of test cases. Followed by T test cases.

Each test case starts with a line containing an integer N ($1 \leq N \leq 10^5$) representing the number of days followed by a line containing N positive integers separated by a single space, representing the list of prices you will assign, each price will be at most 10^6 .

Output

For each test case print a single line containing a single decimal number rounded to exactly 6 decimal places, which is the maximum profit you can get. It's guaranteed that the result will not be more than 10^{18} .

Example

stock.in	standard output
2	6.000000
3	100.000000
1 2 3	
1	
100	

Note

In the first test case, you assign the prices in this order '2 1 3', in the first day you sell the unit you have for 2 dollars, in the second day you buy 2 units for 1 dollar each, and in the third day you sell the 2 units for 3 dollars each and you end up with 6 dollars profit.

In the second test case, you have just 1 day and you must sell the unit you have in that day.