Coding Project

Summaries

Summaries

- **The summary should be a concise substitute for the original problem.**
  - This means the summary must include all key details of the problem.
  - *If you gave a coder your summary without the original problem, they should still be able to arrive at the correct solution.*
- Try reusing **key words** and **notations** from the problem (e.g. “subjects”, “test”, “questions”, “c_i”, etc. rather than synonyms).
- Using pronouns like “You” is fine as long as they remain consistent.
- Keep it **short and to the point**, removing unnecessary data when possible (e.g. names, story details, etc. can be removed).
- A **small paragraph** is more than enough.

✅ Good Summary Example

Problem	Summary
On a beautiful day in December, Daniel invites Andy to play an interesting game: Daniel will pick an array A which has n non-negative integers A_1, A_2,..., A_n and a non-negative integer M. After that, Andy will randomly pick a non-empty sub-array of A, which means he will pick two integers l, r (1≤ l ≤ r ≤ n) và take A_l, A_{l+1},..., A_r from the array A. Andy has to preserve the order of the taken integers. Andy is only allowed to increase his taken integers. Suppose after doing so he receives A'l, A'{l+1},..., A'_r where A'_i ∈ Z,\ A'_i ≥ A_i ∀ i=\overline{l, r}. However, he must follow the restriction below: ∑_{i=l}^{r} A'_i - A_i≤ M. If A'_l, A'_{l+1},..., A'_r in this order form a non-decreasing array, Daniel will award Andy with bubble tea. Right after Andy accepts the challenge, Daniel thinks hard to lower the chance of Andy winning as much as possible. Therefore, when he thinks of array A, Daniel has to calculate quickly the number of non-empty sub-arrays of A that Andy can pick to win this game. After 21/10 second, Daniel has to start the game by telling Andy the array. Please help him to calculate in 2 secs. **Input** Line 1 contains 2 integers n, M. Line 2 contains n integers of array A: A_1, A_2,..., A_n. Numbers on the same line are separated by spaces **Output** An integer indicating the answer	Given an array A of non-negative integers, find the number of subarrays where the minimum number of operations to make it non-decreasing does not exceed M. An operation is defined by picking any value in the subarray and incrementing it by exactly 1.

✅ Good Summary Example

Problem	Summary
E for Easy. Clearly, E for Easy sounds better than D for Deasy, F for Feasy, G for Geasy, or other such nonsense. Perhaps less clear is that E for Easy also sounds better than E for Empossible. It follows that the only logical title for this problem is Easy. Whether this problem is actually easy is up to you. It is 23:16. The relentless rain pelting my window keeps me awake while I write this unintelligible mess. Although my fingers are moving across the k keys of the keyboard and pressing r keys that somehow translate to s complete albeit meaningless words, my mind is elsewhere. The rain outside just made me remember that nightmare I had w days ago. I was chasing my sanity down a network of p dank alleyways of some ghost town in the middle of a desert. My sanity was somewhere ahead of me, just outside of my reach. But this was only the prelude to the nightmare. As I chased my fleeing sanity into a tavern, I came face to face with a 40cm tall messenger pigeon. Frightened out of my wits, I awoke to the sound of birds chirping as if nothing had happened. Unfortunately, my sanity is still hiding inside the tavern. The tavern has n rooms that are connected by m bidirectional halls. The n rooms are uniquely labeled with integers from 1 to n inclusive. The pigeon is in room 1, and my sanity is in room n. Each of the m halls require a certain amount of time to traverse. Even someone like me is able to follow these halls from 1 to n in the minimum amount of time possible. However, there are traps in each hall! Now, each hall only has an interval of time when it is safe to be in it. Should one set foot in a hall when it is unsafe, unthinkable things will happen. I am unable to solve such a difficult problem, so it is up to you to determine the shortest amount of time it takes to go from the room where the pigeon is at to the room where my sanity is hiding. It is now time 0, let the search begin! **Input** The first line contains two space-separated integers n and m (2 ≤ n ≤ 2·10^5;1 ≤ m ≤ 2·10^5), representing the number of rooms and halls respectively. Each of the next m lines contains five space-separated integers a_i, b_i, c_i, d_i, and t_i (1 ≤ a_i, b_i ≤ n;0 ≤ c_i < d_i ≤ 10^9;1 ≤ t_i ≤ 10^9), describing the ith hall. This means the ith hall is between rooms a_i and b_i, it is safe to be in the hall from time c_i to time d_i inclusive and it takes t_i minutes to traverse the hall (in either direction). Two rooms could be connected by multiple halls. It is guaranteed that a_i ≠ b_i. **Output** Output a single integer representing the shortest time it takes to go from room 1 to room n if it is possible to safely reach room n, or output -1 otherwise.	An undirected graph with N nodes and M edges is provided. Each edge has its time interval of [c_i, d_i] during which the edge can be traversed. The traversed edge i takes t_i time. The task is to find the shortest path to reach node 1 to N. If this is not possible, then print -1.

✅ Good Summary Example

Problem	Summary
The mad scientist Dexter performs experiments in molecular chemistry. Today he will try to build a single molecule out of n atoms. It is well-known that a molecule consists of individual atoms, some pairs of which are connected by atomic bonds. Each atom has its value of valence — the number of the bonds the atom can form with other atoms. An atom can form one or several bonds with another atom, but not with itself. The number of the bonds the atom has must be equal to its valency. A molecule must be connected, i.e. for each pair of atoms there must be a path of the bonds that connects the two atoms. Dexter knows the valences of each of the n atoms. Help him find a molecule that can be built out of these atoms, or determine that it is impossible. **Input** The first line of input contains a single integer n (1 ≤ n ≤ 5000), the number of atoms. The second line contains n space-separated integers d_i (1 ≤ d_i ≤ 10^9), where di is equal to the valence of the i-th atom. **Output** If it is possible to build a single connected molecule out of the given atoms, in the first line output “Yes”. Then output n - 1 lines, where i-th line must contain n - i space-separated integer numbers. The j-th number in the i-th line must denote the number of the bonds between the atoms with numbers i and i + j. The atoms are numbered from 1 to n in the order as they appear in the input. If there are several solutions, output any of them. If such molecule does not exist, output “No” in a single line (without quotes).	Given the number of nodes of a graph and their degrees. Find if forming a graph is possible by following the rules below: — Self-loop is not allowed — There can be multiple edges between two nodes — All the nodes should be connected — All the node degrees should match the input degrees.

❌ Bad Summary Example

Problem	Summary	Issues
Hsueh- is about to gradute! Congratulations! Wish all of you a brilliant future like him! Because of Hsueh-'s active performance in the ACM, his gold medals are too many to take home. So he has to choose some of them. Hsueh- has only one box. To make it beautiful, he will stack up his gold medals as a tower, which means every gold medal must be putted on another gold medal except the bottom medal, and there will be exactly one gold medal on each layer. In order to make the tower stable, Hsueh- will choose an integer p, then the lower gold medal should be at least p cm^2 larger than the gold medal right above it. In other words, if the sizes of the tower consists of k gold medals from top to bottom are s_1, s_2, s_3, ..., s_k, then s_1 + p ≤ s_2, s_2 + p ≤ s_3, etc. Hsueh- has n gold medals, the sizes are a_1, a_2, ..., a_n cm^2, and they have the same height. The box can contain at most m(m ≤ n) gold medals. Now Hsueh- was wondering the maximal integer p that can fill the box. **Input** The first line contains one integer T(1 ≤ T ≤ 100), which represents the number of test case. For each test case, the first line consists of two integers n, m(2 ≤ n ≤ 10^3, 2 ≤ m ≤ n) — the number of gold medals Hsueh- has and the number of gold medals his box can contain. The second line contains n space-separated integers a_1, a_2, ..., a_n(1 ≤ a_i ≤ 10^6) — the sizes of n gold medals. **Output** For each test case, print the maximal integer p that can fill the box.	The goal is to find out how many gold medals can be stacked in a box. We are given n number of gold medals and m, the maximum number of medals that can be stacked, while also considering the sizes of each medal. The output required is the maximum number of gold medals that can be stacked in a box.	Key information missing It's critical to explain how the medals' size impact the result.

❌ Bad Summary Example

Problem	Summary	Issues
This is an interactive problem. Kwords has a permutation a_1,a_2,...,a_n of 1 to n, now you can ask him no more than 10 questions to guess the permutation. For each query, you can give him a set s of k(1≤ k≤ n) indexes, he will give you a sorted array of [a_{s_1}, a_{s_2},...,a_{s_k}]. Kwords thinks the problem is very interesting and invites you to solve it with him. **Input** The first line contains a single integer n(1≤ n≤ 1000). **Output** When you are ready to answer, print a single line of the form "! a_1 a_2 ... a_n" (1≤ a_i ≤ n), where a is a permutation.	An unknown permutation a_1,a_2,..., a_n of 1 to 'n' is given. You have to guess the permutation by making no more than 10 queries. For each query, you are given a set of indexes along with a sorted array based on the value of these indexes. Find the permutation.	Too succinct Explaining the query input and its output clearly is important.

❌ Bad Summary Example

Problem	Summary	Issues
There are N cities in Far Far Away. The king wants to connect the N-1 cities with two-way airlines so that: * Each airline connected two different cities. * From each city it was possible to get to any other directly or with transfers. Let's call the accessibility of a city the number of cities with which it is directly connected by airlines. The king also requires that the maximum availability value among all the cities in the kingdom be as high as possible. How many different ways are there to connect cities in the required way? Input data The input contains one integer N — the number of cities (2 ≤ N ≤ 10^9). Output Print a single integer — the answer to the problem.	'n' cities are given and there are 'n-1' roads to connect them to each other. Accessibility of a city is defined as the number of other cities the city is directly connected with a road. Find the number of ways you can connect the cities while maximizing the accessibility of each city.	Imprecise (and very wrong in this instance) We're not trying to maximize the accessibility of each city, but the max accessibility among cities, which is a crucial difference.

Coding Project

Editorials

Editorials

- Explain the **algorithmic family** the solution belongs to and the necessary **high-level steps (+ prominent edge cases)**
    - e.g. dynamic programming, depth-first search, brute force, etc.
- We want to know the reason behind your code
- The Editorial should say “this is how you solve the problem and here’s why”.
- Focus only on the problem’s resolution
- Do **NOT** go line by line like “First do..., then do..., then do...”
- Keep it short and to the point - a small paragraph is more than enough
- The Editorial should be just enough information to convince someone that the proposed method should solve the problem

✅ Good Editorial Example

Problem	Editorial
At the forum, where the problems of Olympiads in Informatics are discussed, the following analogue of captcha was introduced. The participant is given a string of N decimal digits (without leading zeros). As an answer, you need to enter such a base of the number system B, that in this number system the given record will correspond to a composite number (let's call it D), as well as the number X, greater than 1 and less than D, which is a divisor of D. In this case, B and X should not exceed 10^9. Given a string of decimal digits, find any pair of numbers B and X that satisfies the constraints, or answer that there is no solution within the given constraints. **Input data** The input consists of a non-empty string up to 3 · 10^6 characters long, made up of the digits 0 through 9 and not starting with 0. **Output** If there is a solution, print two numbers — the base of the number system B and the divisor of X, written in the decimal number system. Both numbers must satisfy the constraints 2 ≤ B,X ≤ 10^9. If there is no solution, print -1.	If the given number is 1, 2, 3, 5, or 7, then there is no valid base B, the output should be -1. If the given number is 4, 6, or, 8, a valid output is B = 10, x = 2. If the given number is 9, the answer is B = 10, x = 3. If a number is even, one valid solution is B = 10, x = 2. This is valid because in base 10 all the even numbers are divisible by 2. Let's assume S = sum of the digits of n. For the odd numbers the output is: B = S + 1, X = S. This is valid because: X^k - 1 = (X - 1) * (X^(k - 1) + X^(k - 2) + ... + X + 1) The remainder of the number n in the B = S + 1 number system divided by X = S will be 0.

✅ Good Editorial Example

Problem	Editorial
This is an interactive problem. Kwords has a permutation a_1,a_2,...,a_n of 1 to n, now you can ask him no more than 10 questions to guess the permutation. For each query, you can give him a set s of k($1 \le k \le n$) indexes, he will give you a sorted array of [a_{s_1}, a_{s_2},...,a_{s_k}]. Kwords thinks the problem is very interesting and invites you to solve it with him. Input The first line contains a single integer n($1 \le n \le 1000$). Output When you are ready to answer, print a single line of the form "! a_1 a_2 ... a_n" ($1 \le a_i \le n$), where a is a permutation.	The ten queries constraint is a hint that the solution will require either a binary-search, bit-manipulation, or divide-and-conquer techniques. Start by assuming a 0-indexed array and use bits to represent each index. For i-th query, query all indexes for which the i-th bit is on. After getting the sorted array, the required values can be found at the indexes for which the i-th bit is on. Therefore, after making all ten queries, you should have the required permutation.

✅ Good Editorial Example

Problem	Editorial
Haibara is one of Conan's best friends. Conan sent her as a spy in order to discover the killer in one of his investigations. Haibara was on her first mission as a spy. Unfortunately, her mission did not go as planned, and she has been compromised. Haibara got panicked and she forgot what she learned from Conan. So, it is your time to help her to avoid arrest. Your task is to help Haibara find the best safe house for her based on her current situation. There are n safe houses around Haibara's location. The i^th house is d_i kilometers away from Haibara, and it contains m_i coins. According to the manual, the best safe house must be at distance no more than x and must contain at least y coins. If there is more than one safe house satisfying these conditions, you must choose the nearest one. If there is more than one safe house at the same distance, you must choose the one that contains the biggest amount of coins. Finally, if there is more than one safe house satisfying all the previous conditions, you must choose the one with minimum index. Do not forget that Haibara is in a real danger, so you must find the best safe house for her as soon as possible. Conan will be very happy if you rescue Haibara. Can you do this? **Input** The first line contains an integer T (1 ≤ T ≤ 500), in which T is the number of test cases. The first line of each test case contains three integers n, x, and y (1 ≤ n ≤ 500) (1 ≤ x, y ≤ 1000), in which n is the number of safe houses, and x and y are the values from the manual as described in the statement. Then n lines follow, each line contains two integers d_i and m_i (1 ≤ d_i, m_i ≤ 1000), in which d_i is the distance to the i^th safe house, and m_i is the amount of coins it contains. Safe houses are numbered from 1 to n in the order given in the input. **Output** For each test case, print a single line containing - 1 if there is no suitable safe house for Haibara. Otherwise, print the index of the safe house you have chosen for her.	For each house, we store its distance, coins, and index. We iterate through all of the houses while keeping track of the lowest distance so far, the max coins given the house with the lowest distance, and the lowest index given the former two criteria. We use the brute force method that updates the best option once a better option is encountered until there are no more houses to consider.

❌ Bad Editorial Example

Problem	Editorial	Issues
Professor X was concerned about the state of the power generators of the Zapad-1 spacecraft. He did not really like the idea of eating from the energy of the engines, because the path to Mars is not at all close, and anything can happen. What if the fuel runs out on the way back? Astronauts will have to fly all the way to Earth without light, and this is very sad and depressing: neither watch TV, nor read a book, nor listen to music. To solve this difficult problem, the professor decided to power all the electrical equipment of the spacecraft from an independent system of generators purchased from the west (domestic ones were too heavy, like all domestic ones). In the documentation for these generators, it was stated that n generators generate exactly k Joules of electricity in m minutes. The professor knows that for the correct operation of all on-board devices, it is necessary that at least q Joules of energy be received in p minutes. Now he wants to know the smallest number of generators that he needs to install in the spacecraft so that this condition is met. **Input** data The single line of the input file contains five space-separated integers: n, m, k, p, q (1 ≤ n, m, k, p, q ≤ 10^6) — the values described in the problem statement. **Output** The single line of the output file must contain a single integer — the smallest number of generators that need to be installed on the spacecraft.	We could conduct these calculations: genPerMinute = k / m / n and needed = q / p / genPerMinute. Needed would be our final answer, but this approach is risky and may lead to a loss of accuracy. Instead, the formulas can be rewritten so that we replace division with multiplication. Multiplication of up to 3 variables can easily fit in a 64-bit integer. When the formula is rewritten, it looks like this: genPerMinute = k / (n * m) needed = q / (genPerMinute * p) ⟶ q / ((k / (n * m)) * p) ⟶ (q * n * m) / (k * p) This modified formula minimizes the number of division operations and avoids the accuracy loss. We also need to carry out these calculations in a 64-bit integer type and round the result up to the closest integer which is greater or equal.	Too verbose A single sentence explaining why it's a better idea to use multiplications in int64 rather than multiple divisions would have been enough.

❌ Bad Editorial Example

Problem	Editorial	Issues
The mad scientist Dexter performs experiments in molecular chemistry. Today he will try to build a single molecule out of n atoms. It is well-known that a molecule consists of individual atoms, some pairs of which are connected by atomic bonds. Each atom has its value of valence — the number of the bonds the atom can form with other atoms. An atom can form one or several bonds with another atom, but not with itself. The number of the bonds the atom has must be equal to its valency. A molecule must be connected, i.e. for each pair of atoms there must be a path of the bonds that connects the two atoms. Dexter knows the valences of each of the n atoms. Help him find a molecule that can be built out of these atoms, or determine that it is impossible. Input The first line of input contains a single integer n (1 ≤ n ≤ 5000), the number of atoms. The second line contains n space-separated integers d_i (1 ≤ d_i ≤ 10^9), where di is equal to the valence of the i-th atom. Output If it is possible to build a single connected molecule out of the given atoms, in the first line output “Yes”. Then output n - 1 lines, where i-th line must contain n - i space-separated integer numbers. The j-th number in the i-th line must denote the number of the bonds between the atoms with numbers i and i + j. The atoms are numbered from 1 to n in the order as they appear in the input. If there are several solutions, output any of them. If such molecule does not exist, output “No” in a single line (without quotes).	Check if forming a graph is possible. If yes, use any of the priority queue, BST, or segment tree data structures to find the connection between the nodes. Here, the role of a data structure is to find the node with maximum and minimum degrees and connect those values.	Too succinct There’s hardly enough information here to go on. More detail on, eg, how to form the graph would be necessary.

❌ Bad Editorial Example

Problem	Editorial	Issues
Let f(x) be the greatest even divisor of x or 0, if there is no such divisor. You are given n queries li, ri. For each of them you have to find Σ_(l_i ≤ j≤ r_i)f(j). **Input** The first line contains one integer n (1 ≤ n ≤ 10^5) — the number of queries. Each line of n subsequent lines contains two space-separated integers li and ri (1 ≤ l_i ≤ r_i ≤ 10^5) — the description of i-th request. **Output** Print n integers. On i-th line print one integer — the answer for the i-th query.	The code solves the problem by calculating the sum of all even numbers in a given range, then printing the result. The function 'findSumNatural' calculates the sum of all natural numbers up to a given value. The 'findSumEven' function takes two numbers, representing a range of numbers, and calculates the sum of even numbers in the range. Lastly, in the main function, the code takes the test cases as input.	Not contextualizing the answer: why are we summing only the even numbers? Explaining the code directly: we don’t care how the functions themselves are set up.