X - Problem Name

Problemsetters: Max Mustermann

Problem

This is a problem description.

Solution $\mathcal{O}(n)$

This is an explanation of the solution.

A - Attending Classes

Problemsetters: Bernhard

Problem

Given two vertices u, v on a tree calculate the probability that u visits v if you go on a specific random walk.

Solution $\mathcal{O}(n \cdot \log(n) + q \cdot \log(n))$

1. Root the tree at vertex 0 and calculate the probability of a random walk from 0 to all vertices v, call it P_v.
2. Create a data structure to query the lowest common ancestor of two vertices u, v.
3. Let $c = LCA(u, v)$, we can now calculate the probabilty for u, v with P_u, P_v and P_c (With some special cases)
4. You cannot solve each query in linear time, TLE.

B - Broken Polybahn

Problemsetters: Ahmet

Problem

Given a tree, print the number of connected subgraphs with maximum matching of size 1.

Solution $\mathcal{O}(|V|)$

If there is a path with length more than 2, then we have a maximum matching with size more than 1.

B - Broken Polybahn

Problemsetters: Ahmet

Problem

Given a tree, print the number of connected subgraphs with maximum matching of size 1.

Solution $\mathcal{O}(|V|)$

For each vertex v count the number of subgraphs with maximum matching of size 1 it is in it, which is equal to $2^{deg(v)} - 1$.
By inclusion-exclusion answer is $(\sum_{v \in V} 2^{deg(v)}) - 2n - 1$.

B - Broken Polybahn (Challenge)

Problemsetters: Ahmet

Problem

Given a **graph** (V, E), print the number of connected subgraphs with maximum matching of size 1.

Solution $\mathcal{O}(|E| \cdot \sqrt{|E|})$

Left as an exercise.

C - Counting Rectangles

Problemsetters: Justina, Ahmet

Problem

Given a grid with n horizontal and m vertical lines. Count the number of distinct rectangles on the grid.

Solution $\mathcal{O}(1)$

There is a closed formula

$$\binom{n}{2} \cdot \binom{m}{2}$$

D - Dice Game

Problemsetters: Antti

Problem

Expected sum of values of the dice at the end of the game, assuming both players play optimally.

Solution $\mathcal{O}(n)$

Each dice can be handled independently. Do DP on expected value of the dice with k' rounds remaining, assuming the current face is v'. This results in $\mathcal{O}(nk)$ states, $\mathcal{O}(1)$ transition.
You can observe that if k is larger than 100, you can set k to 100(keeping its parity). This results in $\mathcal{O}(n)$ states.

E - Erdős–Ginzburg–Ziv

Problemsetters: Oleksandr

Problem

Given a prime number p and $p-1$ numbers, construct a tree on p vertices, s.t. the sum of edges on the path from 0 to v is $v \bmod p$.

Solution $\mathcal{O}(p \log p)$

- Add edges $x_1, \ldots, x_{p-1}$ one by one.
- Find k s.t. kx_i is connected to 0, and $(k+1)x_i$ isn't.
- Find such k by binary search (no need for monotonicity).
- Start with $L = 0$, $R = ax_i^{-1}$, where a isn't connected yet.

F - Fraudulent Exam

Problemsetters: Bernhard

Problem

Given the a grid of people with different IQs find the largest group that might cheat.

Solution $\mathcal{O}(nm \cdot \log_2^2(nm))$

1. First translate the problem into a graph. All grid position become a vertex and we add edges between adjacent positions in the grid.
2. Let t be a time such that all edges which connect two vertices of between IQs t and $t + k$.
3. Figure out for each edge i the range of times where it is valid $[L_i, r_i]$.
4. Problem now add/remove edges from the graph while maintaining the size of the largest component. Can be done with dynamic connectivity offline.

G - Glitchy Language Model

Problemsetters: Jan, Vincent

Problem

A language model is creating the input and you have to follow the rules closely to parse the input correctly. Then you have to infer values of words given the rules from the input.

Solution $\mathcal{O}(n)$

1. Make sure to follow the description closely when reading in the data.
2. The number of arguments of a function is determined by the number of columns and the number of rows is $< S >^{\text{cols}-1}$.
3. We can always greedily evaluate the functions or queue them until a missing word has a value at a later time.
4. We can not infer value of a word if a word appears in it's subtree in computation graph. (Please don't find fix points!)

H - Hourly Mate

Problemsetters: Oliver, Bernhard

Problem

A language model is creating the input and you have to follow the rules closely to parse the input correctly. Then you have to infer values of words given the rules from the input.

Solution $\mathcal{O}(n \log n)$

1. Observe:
   k of each type possible $\rightarrow k-1$ of each type possible
   k of each type not possible $\rightarrow k+1$ of each type not possible
   $\Rightarrow$ Binary Search
2. Solution will be between 0 and $n/m \Rightarrow \mathcal{O}(\log n)$ checks
3. For a possible k: Take the k Mate of each type with highest expiration date, sort by expiration date, and check if every Mate is not expired when dispensed.
   $\Rightarrow \mathcal{O}(n)$ per check (sort only at beginning)
   $\mathcal{O}(n \log n)$ might also pass (sort in every check)

I - Increased Intelligence

Problemsetters: Bernhard

Problem

Change some allowed letters to maximize the IQ score you get.

Solution $\mathcal{O}(n * \sum_{s \in strings} |s|)$

1. Use Aho-Corasick to create a DFA over the given strings. At each position in the DFA you can calculate the sum of b_i for all substrings i that have been finished.
2. Use dynamic programming over (a, b) where a is the position you are in your DNA sequence and b is the position in the DFA.

J - Just Too Much Procrastination

Problemsetters: Constantin

Problem

To cool a super computer by swapping heat levels, you have to count the minimum number of swaps to maximize the sum of absolute differences of a permutation of n numbers.

Solution $\mathcal{O}(n)$

The maximum sum will always be $\lfloor \frac{n^2}{2} \rfloor - 1$ given by OEIS series A047838.
A permutation maximizes the sum iff. it has a zig-zag pattern, all high numbers are larger than all low numbers and the middle numbers are at the ends. You can try out both middle number configurations with a linear scan and take the minimum.

K - Keen on Rösti

Problemsetters: Ahmet

Problem

Expected amount of loss when you start at k-th place in the queue of n people. Every round you pay 1 franc with probability $1 - p$.

Solution $\mathcal{O}(1)$

Let $q = 1 - p$. Let X_i represent the indicator random variable indicating whether the k-th student pays his i-th 1 franc (during i-th round).

$E[X_1 + X_2 + X_3 + ...] = E[X_1] + E[X_2] + E[X_3] + ... =$
$q^k + q^k * q^n + q^k * q^{2n} + ... = q^k \cdot (\frac{1}{1-q^n})$

The answer is $q^k \cdot (\frac{1}{1-q^n})$.

L - Locomotive Control Center

Problemsetters: Mihnea, Ahmet

Problem

At **station A**, there are n railcars, numbered with distinct values from *1 to n*, in an arbitrary order. However, the order in which they are in track A is important, as they *will leave point A exactly in the order in which they are listed*. Using **station B** as a stack and having access to operations A-B, A-C, B-C, is it possible to move the railcars from A to C in descending order? Find the **shortest** sequence of operations

Solution $\mathcal{O}(n)$

- Use station B as a stack
- Prioritize the A-C type operations

L - Locomotive Control Center

Problemsetters: Mihnea, Ahmet

Problem

- When should we use station B?
- Is the sequence unique?
- What about the shortest one?
- Is the time complexity actually O(n)?

Solution $\mathcal{O}(n)$

Use station B as a stack