IDI Open 2024

Presentation of solutions

March 16th 2024

The Judges and Testers

Judges

- Jean Niklas L'orange (Kodemaker)
- Tobias Meyer Andersen (SINTEF Digital)

Test Solver

- Johan Sokrates Wind (University of Oslo)

We need more judges and testers! Send an email to `jeannikl@hypirion.com`, or contact any of the judges/organizers after the presentation.

Deceitful Dice

- Sort the dice in descending order of expected value
- Make sure the sorting is stable!
- Either sort pairs, or use explicitly stable sorting algorithm

Solved by XXXX teams
First solution after 2 minutes

Exponential Eggs

- Implement the formula, e.g.
- NB: Since output is asked as integer, print the entire number as an integer

Solved by XXXX teams
First solution after 5 minutes

Friend Finding

- You must figure out how quickly they can meet when placed at different nodes in the graph
- You do not need to consider arbitrary triples of nodes, because nodes on the same path must be strictly better
- You do not need to consider paths of length greater than 3, because placing the just next to each other and meeting in the middle is strictly better
- Answer is the length of the second shortest edge of any node

Solved by XXXXX teams
First solution after 28 minutes

Puzzling Poison Protection

- All items are initiall marked as unknown (?)
- First, mark all items in a record that doesn't protect against poison as `n`
- Then, go through all records that protects against poison:
  - If there's only one item that isn't marked with `n`, mark that item as `y`

Solved by XXXX teams
First solution after 45 minutes

Möbius Music

- Given a sequence, and a rotation of it, find out how much it was rotated by
- This can be phrased as matching pattern in a text if you consider the text (our music piece) to be a cyclical string!
- If remembering or implementing KMP on the spot is hard, use a rolling polynomial hash to do it probabilistically.

Solved by XXXX teams
First solution after 14 minutes

Climbing Capacity

- Agnar minimizes T, the level of tiredness on the most tiresome day, while finishing the hike within M days or less
- Observation I) Evaluating IsPossible(T) for all possible values gives a sorted array
- Observation II) Given a maximum level of tiredness we can greedily evaluate if it is possible in linear time
- This means we can binary search over T
- To make it fast enough avoid multiplication of large numbers using the logarithm rule $log(A * B) = log(A) + log(B)$

Solved by XXXXX teams
First solution after 28 minutes

Terminal Tools

- Bipartite matching problem between features and characters
- Solved by computing the maximum flow of the graph connecting a source to each feature and every character to a sink.
- Nice restrictions should make it solvable with any maximum flow algorithm of your liking

Solved by XXXX teams
First solution after 33 minutes

Restaurant Rinser

- Use dynamic programming to find the total amount of possible paths ending at a certain location of every possible length t

$$dp(t, x, y) = \Sigma dp(t - 1, neighbors)$$

- Redo it without counting paths go to or from the dirty table, we now have number of paths with dirty table
- Subtracting number of paths with the table dirty from the number of possible paths leaves us with number of paths cleaning the table
- Computing multiplicative inverse mod prime is done using Fermat's little theorem adjusted slightly

$$n^{p-1} \equiv 1 \pmod{p} \rightarrow n^{n-2} \equiv n^{-1} \pmod{p}$$

- The fraction is then just a regular multiplication

Solved by XXXXX teams
First solution after 61 minutes

Sausage Slicing

- Divide a string of sausages fairly, everyone must get a fair share from their persepctive of evaluations
- Dynamic programming to figure out how many people of type A and B can be satisfied by the first i sausages.
- $dp(i, a, b) =$ true iff a people of type A and b people of type B can receive a fair share using only the first i sausages
- if a range of sausages from $i + 1$ to j is considered a fair portion by a person from group A:

$$dp(j, a, b)| = dp(i, a - 1, b)$$

- precompute prefix sums to check this faster

Sausage Slicing

- This is the intended solution, limits based on $\mathcal{O}(N^2 \cdot A \cdot B + N)$
- Johan found valid solution in $\mathcal{O}(A \cdot B + N)$, so limits on N, A, and B could have been pushed much higher...

Solved by XXXX teams
First solution after 68 minutes