Solutions – IDI Open 2018

April 14^{th} 2018

Boat Parts, Author: Jean Niklas L'orange

- Make an empty set of boat parts (strings)
- For each day, put the day's boat part into the set
- If the size of the set is equal to the number of parts, print out the day
- If we reach the end, print `paradox avoided`

Solved by 53 teams
First solution after 2 minutes

Cross Country, Author: Karl Johan Sande Heimark

- A basic single source shortest path problem
- The constraints are loose enough to make it possible to use Floyd-Warshall

Solved by 39 teams
First solution after 6 minutes

Gear Changing, Author: Karl Johan Sande Heimark

- Find the ratio between the crank and the back wheel for all the possible gears.
- Sort them!
- Compare each one to the next and make sure the ratio between them is $< P\%$.

Solved by 30 teams
First solution after 16 minutes

Gruesome Cave, Author: Jean Niklas L'orange

- Limits allows us to use Markov chains to find probability of a Grue to be in a location
- Use Matrix exponentiation to find the converging result. Be careful to handle odd/even properly.
- Or you realise that $P_{i,j} = \frac{\text{edges}_{i,j}}{\text{total number of edges}}$
- When probabilities are found, use Dijkstra from entrance to diamond

Solved by 18 teams
First solution after 60 minutes

Witchwood, Author: Jean Niklas L'orange

- Key insight: Placing log n in the least amount of time means you also have to place log $n - 1$ in the least amount of time
- Let $E(X_i)$ be the expected optimal time to place log i ($E(X_0) = 0$). We then have

$$E(X_i) = \arg\min_j E(X_{i-1}) + T_j + P_j(E(X_i) + K)$$

- Solved, this yields $E(X_i) = \arg\min_j \frac{T_j + E(X_{i-1}) + P_j K}{1 - P_j}$
- Trying all possible j for all $E(X_i)$ gives $\mathcal{O}(NM)$ solution

Solved by 11 teams
First solution after 94 minutes

Secret Santa Cycles, Author: Torbjørn Morland

- First identify the interesting features of the graph
- Some people get no gifts
- Some people get multiple gifts
- Some cycles might be isolated from others

Solved by 7 teams
First solution after 137 minutes

Secret Santa Cycles, Author: Torbjørn Morland

- Number of people without gifts == number of extra gifts to others
- If you redistribute these, you end up with a set of cycles
- Instead of changing to a person with no incoming gifts, give it to someone in another cycle
- Answer is number of people with no incoming gifts + number of isolated cycles

When Planetoids Align, Author: Jean Niklas L'orange

- First sort planetoids by distance from Sol
- Then make a segment tree where leaf nodes are the the planetoids
- Define all other nodes as a subspace, where its orbital period equals to `period = lcm(left.period, right.period)`
- Querying the tree for subspaces that cover an area will give you at most $\mathcal{O}(\log n)$ subspaces, which you can perform `lcm` on
- **NB**: *next day* means empty regions should output 1, not 0

Solved by 6 teams
First solution after 135 minutes

Cuboid Slicing Game, Author: Eirik Reksten

- Impartial game, i.e. game with perfect information, where same move is available to both players in a given game state
- Impartial games can be solved by tree search. A state is winning if there is a move to a losing state, a state is losing if all moves lead to winning state
- However, the game tree would be infeasibly large in this game
- Consider the cuboids in a state as separate game, and somehow determine if the collection of games is winning or losing

Solved by 1 team
First solution after 164 minutes

Cuboid Slicing Game, Author: Eirik Reksten

- Equivalent to Nim, by the Sprague-Grundy theorem
- According to the Sprague–Grundy theorem, the nim-value of a game position is the minimum excluded value of the class of values of the positions that can be reached in a single move from the given position
- Solve with DP or depth first with memoization
- If the value of the starting game state is zero, it's a losing state and the opposite player wins. If it's non-zero, the current player (whose name is given in the input) wins

Number Anagrams, Author: Torbjørn Morland

- First generate all the numbers, discarding the duplicates
- All numbers with the same digits can transform to the same numbers
- Represent a number as a string with the digits in sorted order
- Count how many different numbers exist for each of these

Solved by 1 team
First solution after 285 minutes

Number Anagrams, Author: Torbjørn Morland

- Make a tree where the children are the transformed numbers you can reach from that node
- Search the tree for the longest number you can make, memoizing the length and result for each node
- For a given node, find the number of different numbers that transform to the node, a, the maximum possible length for all children, l. Result for a node is a * sum(children that have max length l).
- If it has no children, the answer is 1.
- Remember to use modulo on all operations

Draft Time, Author: Karl Johan Sande Heimark

- Start by letting players choose their favourite team.
- For player P go through the list of teams in prioritized order:
- If the team has a spot, add P to the team and continue with next player.
- If the team is full, compare P to the player T: the player the team likes the least of the ones they already have.
- If the team would rather have T than P continue with next team on Ps list.
- If the team would rather have P than T add P to the team and add T to the ones needing a team.
- Continue until solved for all players.

Solved by 0 teams

Polygon Game, Author: Karl Johan Sande Heimark

- Problem: Divide polygons into smaller polygons and find the area of the largest one remaining
- Easy to solve, but hard to program correctly.
- To cut a polygon by a line L and keep the left side:
- For point P_i in Polygon
- If P_i is on the left side of L, keep P (cross product).
- If P_i and P_{i+1} are on different sides of the line, find the intersection and keep that.
- You can then simply reverse the line to find the right side as well.

Solved by 0 teams

Picture Day, Author: Piotr Zielinski, Torbjørn Morland

- Look at two groups A_i and B_j
- The cost for taking pictures of everyone in $A_i \cap B_i$ is $\min(a, b, c * |A_i \cap B_i|)$
- For two groups A_i and B_j, make an edge between the nodes with weight c * (number of students in the intersection of A_i and B_i).
- Make two special nodes S, T
- Make edges from S to all A_i with weight a.
- Make edges from ann B_i to T.

Solved by 0 teams

Picture Day, Author: Piotr Zielinski, Torbjørn Morland

- Make a graph where each group from A and B are nodes
- For two groups A_i and B_j, make an edge between the nodes with weight $c * |A_i \cap B_i|$.
- Make two special nodes S, T
- Make edges from S to all A_i with weight a.
- Make edges from all B_i to T.

Picture Day, Author: Piotr Zielinski, Torbjørn Morland

- Note that all paths from S to T has exactly 3 edges, corresponding to the alternatives for taking pictures of $A_i \cap B_i$
- Answer is min cut of this graph
- Min cut == max flow, so can be solved with your favorite max flow algorithm