

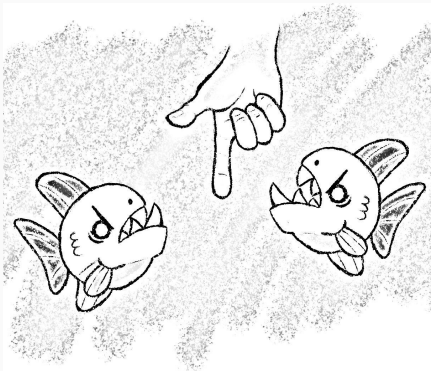
UKIEPC 2020



Summary and solution outlines

Problem Solutions

Note: Many problems this year are shared with BAPC (<https://2020.bapc.eu/>)



Aquarium

?? correct • solved at: ??:?? by

??

University of ??

Author: **Freek Henstra**

Overview

- We need to line some piranhas up for a group photo.
- Putting a finger in the tank causes adjacent piranhas to start swimming toward the finger.
- Find the number of seconds of finger exposure needed to put every fish in its specified place.

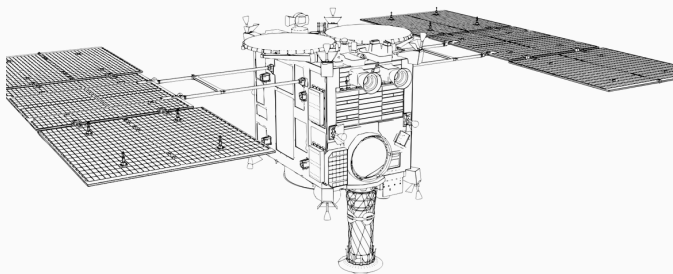
Aquarium - Solution

Techniques

- Intervals
- Greedy algorithms
- Amortisation

Algorithm

- Each fish needs to move by X_i places.
 - The number of finger placements to the right must be X_i more than the number to the left.
- Assume the number of placements to the left of the leftmost fish is $-X_0$.
 - Then the next number must be 0 , the next after that must be $0+X_1$, then $0+X_1+X_2$, etcetera.
 - We need all values to be non-negative. Take the minimum of value and subtract it from everything.
- Now we need to check if this number of placements is feasible
 - The answer is guaranteed to be at most $O(N \cdot K^2)$.
 - So we can just go from left to right making all valid moves.



Bidirectional

?? correct • solved at: ??:?? by
??
??

Authors:
Ludo Pulles, Mike de Vries

Overview

- Write a number $< 10^{18}$ as sum of ≤ 10 palindromic numbers.

Bidirectional Code - Solution

Techniques

- Greedy
- Recursion
- Stubbornness

Algorithm

- Idea: construct the biggest balanced number less than n greedily.
- Example:
 - $n = 970\ 894\ 988\ 875\ 162\ 603$
 - $p[1] = 970\ 894\ 987\ 789\ 498\ 079$
 - $n - p[1] = 000\ 000\ 001\ 085\ 664\ 524$
 - $p[2] = 000\ 000\ 001\ 085\ 555\ 801$
 - $n - p[1] - p[2] = 000\ 000\ 000\ 000\ 108\ 723$
- Possible edge case: $n = 10^k : p = n - 1$.
- To get $k \leq 5$: run a brute force to express n as sum of three balanced numbers when $n \leq 200\ 000$.

United Kingdom and Ireland programming contest

RANK	TEAM	SCORE	A	B	C	D	E	F
1	Prime Goats	140						
2	Bananas Tony	75						
3	Bubbly Hair	101						
4	Konopora Predplatena	120						
5	AmatRINciana	30						
6	Spare Camera	70						
7	Use A Camel	200						

- Reconstruct a missing part of a contest scoreboard from other known parts.

Corrupted Judge

?? correct • solved at: ??:?? by
 ??
 ??

Author: Boas Kluiving

Corrupted Judge- Solution

Techniques

- Logic
- Implementation
- Zalgo

Algorithm

- Starting at the last team fill in the corrupted column conservatively.
The last team solved $p[n]=1$ problem.
 - For $i=n-1, \dots, 1$ $p[i] = p[i+1] + (1 \text{ if } t[i] \leq t[i+1] \text{ else } 0)$
 - If $p[1] = p$, then the scoreboard is non-ambiguous.
 - Else $p[1] = p$ gives another correct scoreboard (ambiguous)
- Some special cases to deal with
 - If no team has solved any problem, it is not ambiguous.



Divvying Up

?? correct • solved at: ??:?? by
??
??

Author: **Robin Lee**

Overview

- Three professionals (one not pictured) are going to earn some money from individual events
- Can they distribute the money equally?

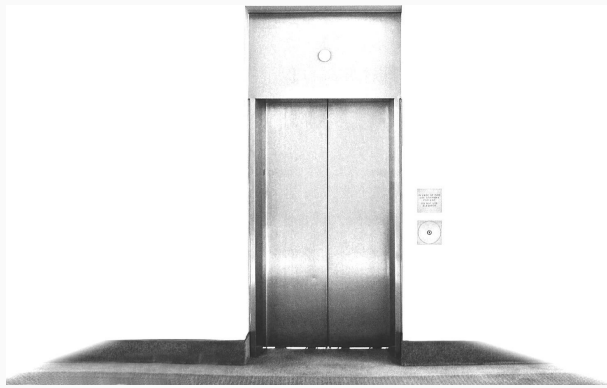
Divvying Up - Solution

Techniques

- Integers

Algorithm

- We just need to know if the total is divisible by 3. So start by computing the total as the sum of the input array.
 - In python, `sum(map(int, input().split()))` does the job.
- Use your language's modulus operator (usually "%") to calculate the remainder from division.



Elevator Pitch

?? correct • solved at: ??:?? by
??
??

Author:
Mees de Vries

Overview

- Build the least number of elevators so that all buildings become accessible.
- We need to count local maxima in the floor plan, but only once each.

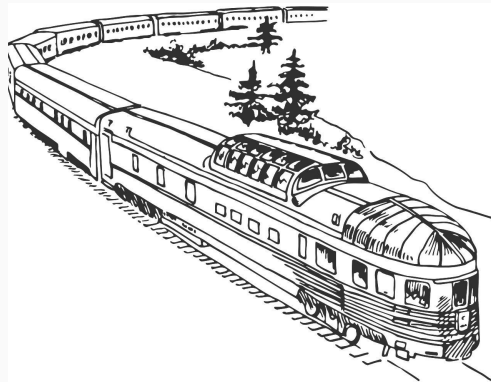
Elevator Pitch - Solution

Techniques

- Breadth-first search
- Implicit graphs

Algorithm

- Sort all of the locations in the grid by (descending) height
- Go through all of the locations in this order:
 - If the location is already visited, skip it.
 - Otherwise, run a flood fill (BFS) from that location to all lower/equal locations, and mark them as visited.
- Output the total number of flood fills needed



Family Fares

?? correct • solved at: ??:?? by
??
??

Author: **Boas Kluiving**

Overview

- A number of people are travelling to the same place along shortest paths, and we can buddy some of them up with a group ticket.
- Group tickets cost a fixed amount per person and must be taken together.
- Which group ticket starting point should we choose (if we choose one at all)?

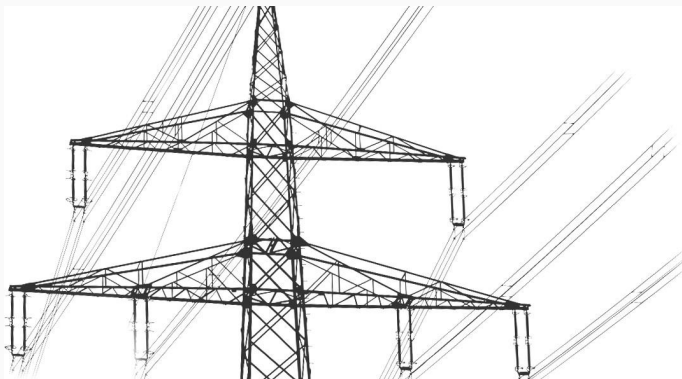
Family Fares - Solution

Techniques

- Dijkstra
- Memoisation
- Bitmasks

Algorithm

- We need to precompute a lot of shortest paths to the idyllic village (which is where the group ticket will always end up).
- For every family member mark all stations which are on shortest paths from their starting point to the idyllic village.
 - Optimise for higher numbers of people by using a bitset and ORing it with shortest-path neighbours.
- Iterate over all stations and compute the savings when using that station to buy the group ticket.



Generators

? correct • solved at: ??:?? by
??
??

Author: **Timon Knigge**

Overview

- Install some power plants onto an island where everybody “lives on the edge” of a giant weighted cycle.
- To do this, pick some edges and vertices such that each vertex is connected to a marked vertex via a direct or indirect path.

Generators - Solution

Techniques

- Spanning Trees

Algorithm

- Transform it into a “minimum spanning tree” problem where every node has to be connected to a virtual “power” node.
- We’ll put this power node at the centre and connect it to everything else with a weight according to the cost of installing a power station there.
- Prim’s or Kruskal’s algorithms can do the rest.



Haughty Cuisine

?? correct • solved at: ??:?? by
??
??

Authors:

Pim Spelier, Mike de Vries, Ragnar Groot
Koerkamp, Robin Lee

Overview

- Solve an annoyance by picking a list of food items from a list of set menus at random.

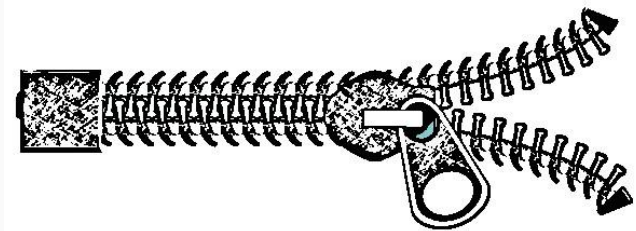
Haughty Cuisine - Solution

Techniques

- Implementation

Algorithm

- Read in the size of the first menu, and immediately print it back out again.



Incomplete Sort

?? correct • solved at: ??:?? by
??
??

Author: **Jorke de Vlas**

Overview

- Sort an array by sorting half of it three times.

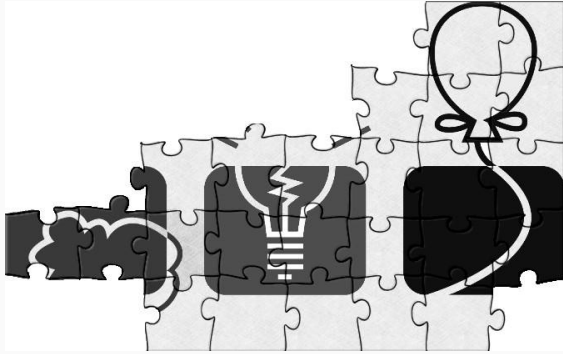
Incomplete Sort - Solution

Techniques

- Divide and conquer
- Permutations

Algorithm

- Idea:
 - In the first step, sort the first quarter.
 - In the second step, sort the second quarter.
 - In the final step, sort the remaining numbers.
- To accomplish this:
 - First step: Choose the first $n/4$ numbers and the positions of the first $n/4$ numbers, so that the first $n/4$ numbers are forced into the first quarter.
 - Second step: Choose the next $n/4$ numbers and positions, for the same reason.
 - Third step: Just sort everything else ($n/2$)



Jigsaw

?? correct • solved at: ??:?? by
??
??

Author: **Mike de Vries**

Overview

- Determine whether a jigsaw puzzle is solvable from the numbers of each kind of piece.

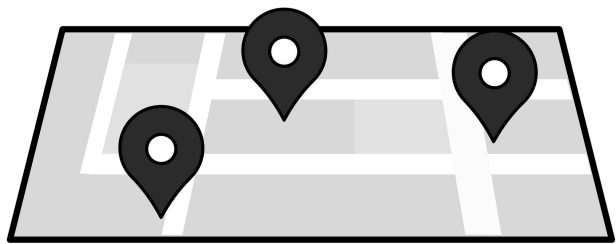
Jigsaw - Solution

Techniques

- Ad Hoc

Algorithm

- A jigsaw puzzle of size $w \cdot h$ contains:
 - 4 corner pieces
 - $2(h - 2) + 2(w - 2)$ edge pieces
 - $(h - 2)(w - 2)$ center pieces
- This reduces the problem to a simple system of equations.



Kleptocrat

? correct • solved at: ??:?? by
??
??

Author: **Jorke de Vlas**

Overview

- Given a connected undirected graph, find a path from A to N that minimizes XOR of the values on the edges.

Kleptocrat - Solution

Techniques

- Linear Algebra
- Cycles

Algorithm

- Observation: walking back and forth does not change the XOR-value since $x \oplus x = 0$.
- When there is a cycle starting from c with XOR-value v , we may walk from a to c , around the cycle, back to a and then to b giving a value of $w \oplus v$ where w was the value of a path from a to b .
- This is an equation over \mathbb{F}_2 and the v_i can be reduced with Gaussian Elimination giving 64 values.

Kleptocrat - Solution (cont.)

Techniques

- Linear Algebra
- Cycles

Algorithm

- See v_i as vectors in F_2^{64} by writing v_i in base 2.
- The linear combinations form a subspace of dimension at most 64: find a basis, which has at most 64 elements.
- Now, given an initial path w , for i from 63 to 0, look if w has a 1 in the i th binary digit and check if there is a basis element with i as most significant digit, in which XOR w with this value.



Lost Map

?? correct • solved at: ??:?? by
??
??

Author: **Robin Lee**

Overview

- Two map fragments (strings) are badly damaged and some items are replaced by “?”s
- Find how many ways you can overlay the two fragments so that no non-“?” values conflict.

Lost Map - Solution

Techniques

- FFT
- Convolution

Algorithm

- We can treat the “?” and non-“?” values completely separately.
 - We make two separate binary strings
 - One encoding “?” as 0 values and everything else as 1 values
 - One encoding “?” as several -1s and everything else as its binary representation (takes $\log K$ bits where K is the number of directions)
 - If we reverse one array, then the **convolution** of both versions of the binary strings from each map can be used to find the number of bits matching each time
 - We need the number of bits across both convolutions to exactly equal the length of the string.
- Pitfalls: FFT needs to be **fast**.



Moderate Pace

?? correct • solved at: ??:?? by
??
??

Author: **Bergur Snorrason**

Overview

- Print the median value at the same index of three different arrays.

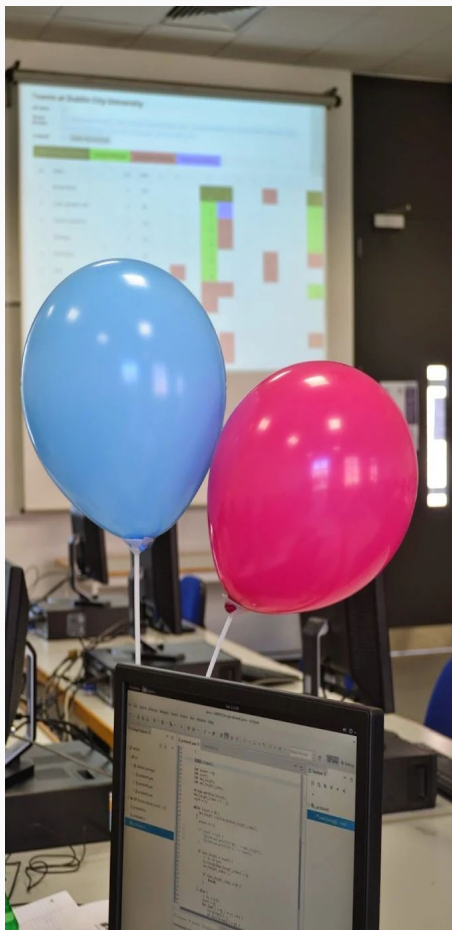
Moderate Pace - Solution

Techniques

- Sorting
- Implementation

Algorithm

- Read in the three arrays and iterate from $i=0$ to $i=n-1$
 - Create a new list L with the three items at index i and sort it
 - Print $L[1]$, and move onto the next.



Questions?

Or comments?

Final Standings

<http://domjudge.bath.ac.uk/>

