

Santa Claus and his socks puzzle sheet

The problem

Santa Claus's dirty socks

It's almost midnight on the December 24th and there was a lot of activity on the North Pole. The little elves were wrapping up the last 64 toys for good boys and girls. The elves then call Santa Claus and tell him: "Everything is ready!". Santa Claus is thrilled and eager to go in his sleigh, but Mrs Claus enters and asks "Where are your dirty socks, Santa Claus?". And then an elf remembers that they went into one of the 64 boxes!

Quick, quick, we need to find those socks, and as soon as possible, the children won't wait!

The socks are quite heavy so it is possible to detect them in a box if we weight it on a scale against another box. Luckily, the elves have such a scale handy.

How are the elves going to find the socks as quickly as possible?

How long will it take?

Data

There is only one scale.

There are 64 parcels (level 1).

There are 128 parcels (level 2).

There are 256 parcels (level 3).

There are 1024 parcels (level 4).

There are 65 parcels (level 5).

The socks are in one of them.

A scale is the only tool.

The scale can weigh any number of parcels against any other number.

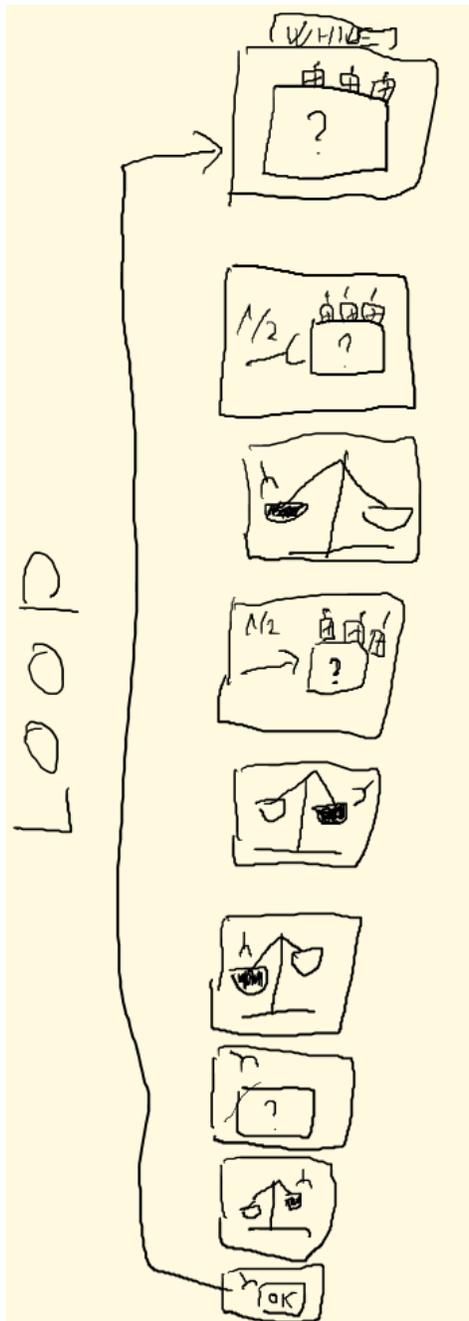
We have two zones on the screen:

- To be tested (zone 1)
- Already tested and without the socks (zone 2)

Solution

This problem is a typical divide-and-conquer one, with a very simple algorithm.

1. Graphical



2. Algorithmical

General solution ($N=2^n$)

It's a simple case of dividing by two the number of parcels and comparing them. We discard the lighter one and continue up until there is only one left, which is the heaviest.

While (there is more than one parcel in zone 1)

Do {

Pick all the parcels in zone 1

Put half of the parcels on one of the pan

Put the other half on the other pan

Weigh (automatic process)

Put the heavier half in zone 1

Put the lighter half in zone 2

}

The parcel remaining in zone 1 is the one we are looking for

The less trivial solution ($N \neq 2^n$)

Here we have the additional issue of having uneven numbers. The simplest way is to be sure we are only dealing with even number, leaving any additional parcel in zone 1.

While (there is more than one parcel in zone 1)

Do {

Pick all the parcels in zone 1

If there is an uneven number

Then leave 1 in zone 1

Put half of the parcels on one of the pan

Put the other half on the other pan

Weigh (automatic process)

Put the heavier half in zone 1

Put the lighter half in zone 2

}

The parcel remaining in zone 1 is the one we are looking for

List of needed assets (aka talking about the assets)

1. Action

We need to be able to pick some parcels, to put them on the first or second pan, to weigh them, to discard them or to put them back in the testing zone

- A1. Robot goes left (existing)
- A2. Robot goes right (existing)
- A3. Robot picks one parcel from zone 1 (new)
- A4. Robot picks half the parcels from zone 1 (new)
- A5. Robot throws parcels in zone 1 (new)
- A6. Robot throws parcels in zone 2 (new)
- A7. Robot puts what he has in the left pan (new)
- A8. Robot puts what he has in the right pan (new)
- A9. Robot picks from the heaviest pan (new)
- A10. Robot picks from the lightest pan (new)

2. Tests

- T1. There are a lot of parcels in zone 1 (new)
- T2. There is more only one parcel in zone 1 (new)
- T3. There is no parcel in zone 1 (new)
- T4. There are a lot of parcels in zone 2 (new)
- T5. There is more only one parcel in zone 2 (new)
- T6. There is no parcel in zone 2 (new)

3. Code

C1. IF-THEN-ELSE (existing)

C2. WHILE (existing)

4. Result zone general purpose assets

GP1. Robot (existing)

GP2. Empty zone 1 (new)

GP3. Half-full zone 1 (new)

GP4. Full zone 1 (new)

GP5. Empty zone 2 (new)

GP6. Half-full zone 2 (new)

GP7. Full zone 2 (new)

GP8. Scale (balanced) (new)

GP9. Scale tilting to the left (new)

GP10. Scale tilting to the right (new)

GP11. Empty left pan (new)

GP12. Full left pan (new)

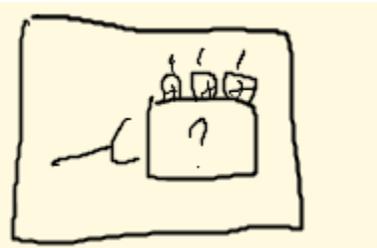
GP13. Empty right pan (new)

GP14. Full right pan (new)

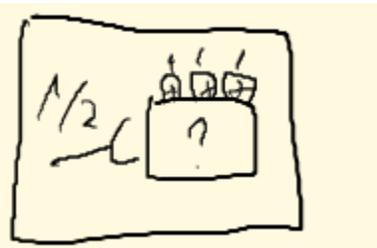
Visualization (aka drawing the assets)

1. Action

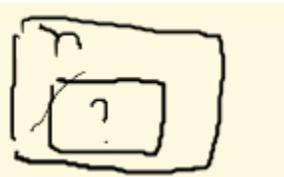
A3. Robot picks one parcel from zone 1 (new)



A4. Robot picks half the parcels from zone 1 (new)



A5. Robot throws parcels in zone 1 (new)



A6. Robot throws parcels in zone 2 (new)



A7. Robot puts what he has in the left pan (new)



A8. Robot puts what he has in the right pan (new)



A9. Robot picks from the heaviest pan (new)

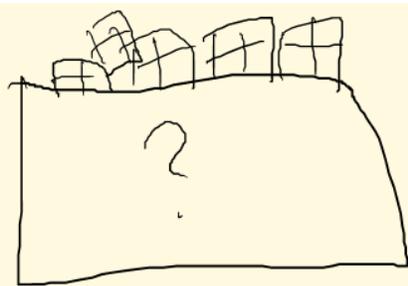


A10. Robot picks from the lightest pan (new)

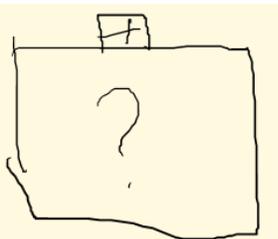


1. Test

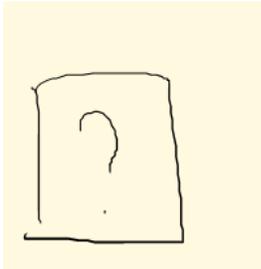
T1. There are a lot of parcels in zone 1 (new)



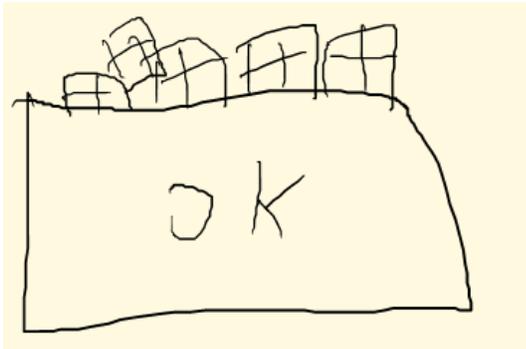
T2. There is more only one parcel in zone 1 (new)



T3. There is no parcel in zone 1 (new)



T4. There are a lot of parcels in zone 2 (new)



T5. There is more only one parcel in zone 2 (new)

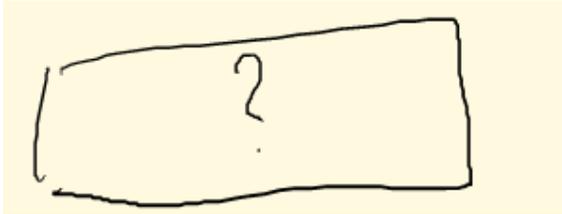


T6. There is no parcel in zone 2 (new)

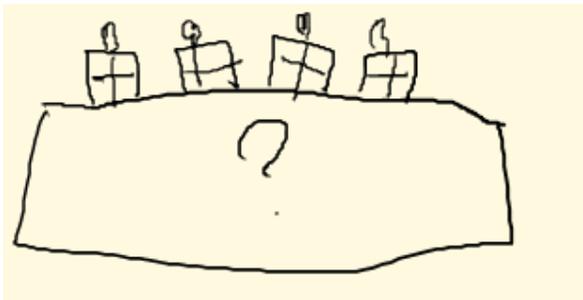


1. General purpose

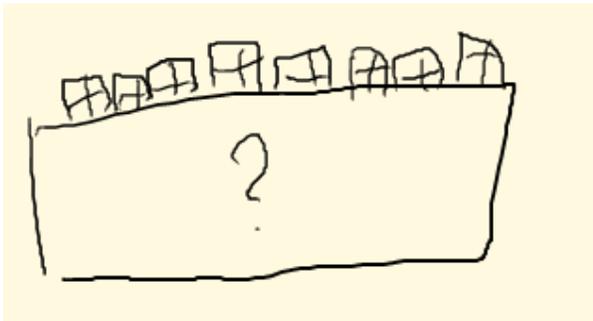
GP2. Empty zone 1 (new)



GP3. Half-full zone 1 (new)



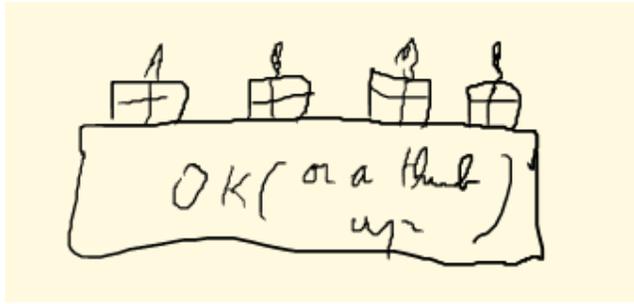
GP4. Full zone 1 (new)



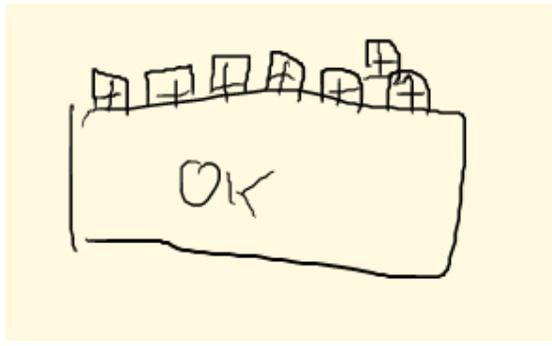
GP5. Empty zone 2 (new)



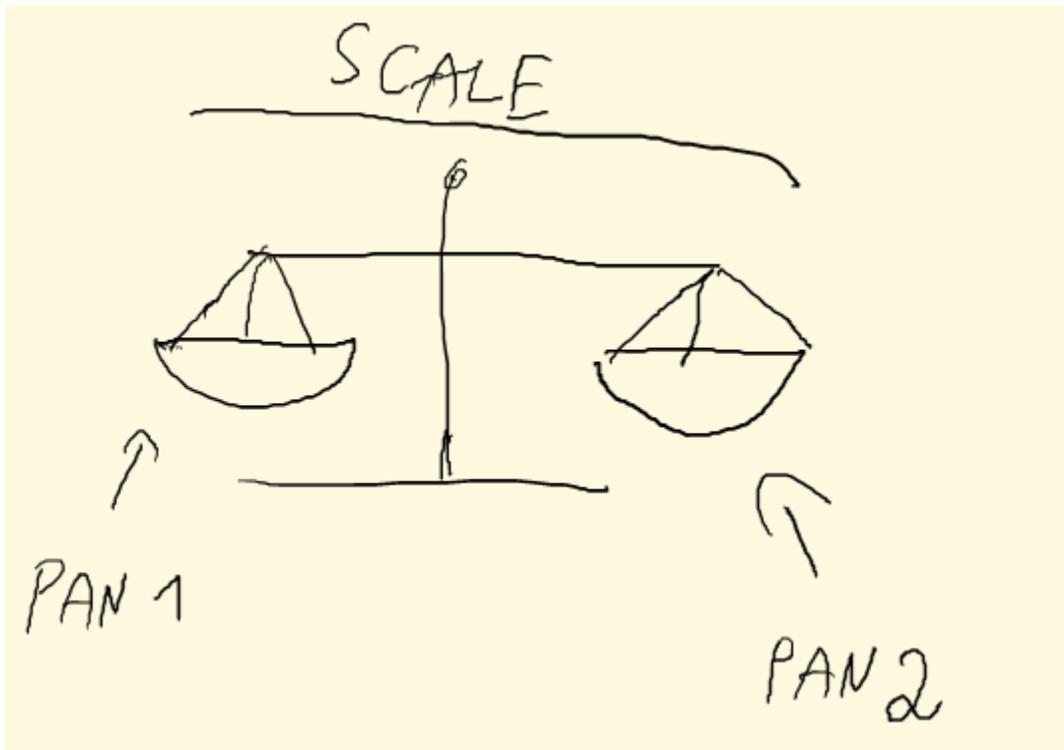
GP6. Half-full zone 2 (new)



GP7. Full zone 2 (new)

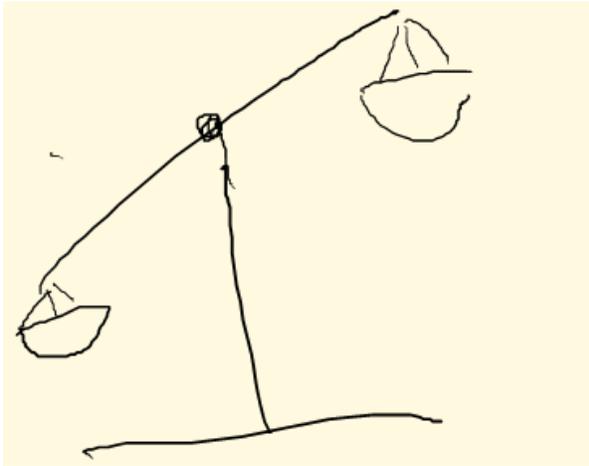


GP8. Scale (balanced) (new)

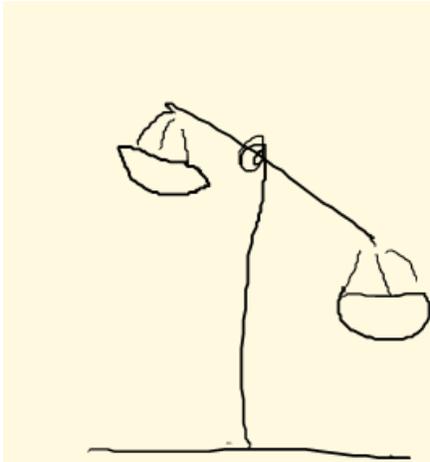


GP9. Scale tilting to the left (new)

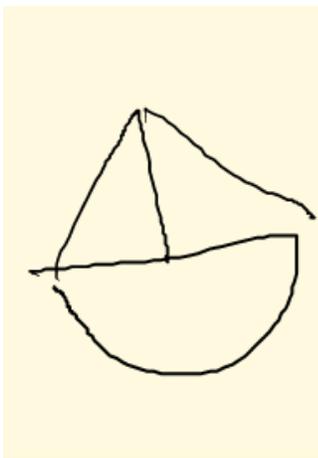
I would suggest differentiating the left and right pan of the scale by a color or any other graphical sign.



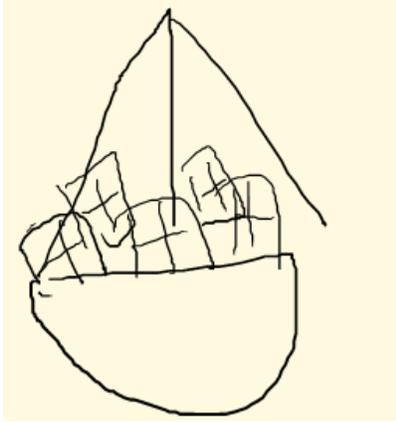
GP10. Scale tilting to the right (new)



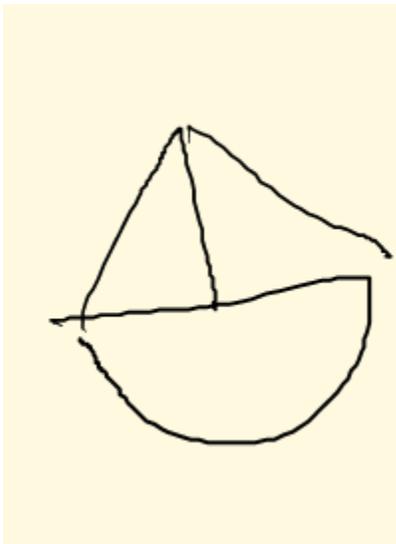
GP11. Empty left pan (new)



GP12. Full left pan (new)



GP13. Empty right pan (new)



GP14. Full right pan (new)

