

Accountable tech interview challenge.

Welcome to Corg, a small, highly regulated country in the middle of nowhere. The inhabitants of Corg like to trade items, but they are highly regulated on what they can trade.

You are asked by the council of Corg to create a system to facilitate the regulated trading.

The inhabitants Corg trade in *coal (in kg)*, *cheese (in kg)*, *books (in quantity)* and *bikes (in quantity)*, their local currency is *Corg Gold (cg)* written as *cg 4,12*.

The regulations:

general buying limits

- No single inhabitant can own more than 10% of the global coal market
- No single inhabitant can own more than 2 bikes
- you can't buy bikes twice in a row
- you can't sell books 3 times in a row

taxes

On each trade, there is a certain amount of tax that has to be payed to the government of Corg, this gets deposited directly into the government bank account.

Buy taxes, when you buy something, you have to pay an extra tax

- coal: 75%
- cheese, bikes: 15%
- books: 0%

Sell taxes, when you sell something, you have to pay taxes

- cheese, bikes, books: 5%
- coal: 15%

tax for having items

- For each bike an inhabitant owns, coal has a *-5%* sell tax (e.g 1 bike = 10% tax / 2 bikes = 5% tax / 3 bikes = 0% tax) to a limit of 0%
- for each kg of coal you own, buying books is taxed 5% more (e.g. 1kg of coal = 5%, 2kg = 10%) to a limit of 50%

Birth

When a Corg citizen is born, they will start life with resources and *cg*, their life is about trading so they don't have to work for their start capital.

How trading works

You can sell items for **cg** and you can buy items for **cg**. You can not directly trade items.

Example of the flow:

1. inh.[a] sets up a sell offer for **12kg of coal** for **cg 16,00**.
2. inh.[b] accepts the offer
3. inh.[a] pays sell taxes, inh.[b] pays buy taxes
4. **cg 16,00** gets removed from inh.[b] and inserted into inh.[a] & **12kg coal** gets removed from inh.[a] and inserted into inh.[b]

deliverables:

functional

Corg is a transparent country, and the inhabitants should be able to see what's in the governments bank account at all times.

- an endpoint to check the goverment bank account

To be able to accept an offer, you'd need to see the list of offers

- an endpoint to view all offers that are still open

To be able to list offers, you'd need to be able to create one

- an endpoint to create a new offer

Offers are useless if you can't accept them

- an endpoint to accept an offer

technical

- Node.js (ts/js)
- Basic testing
- An API (Koa/Express/...)
- Authentication is not required
- You can store data in memory (or db/files)
- No front-end is required

Assumptions

It might be possible that you have to make some assumptions along the way, please document those assumptions.

Have fun! And please make sure to send over any questions when you have them.