Backend Challenge

The purpose of this challenge is to test your ability to write a clean and simple application that is manipulating some configuration data coming from a fictional frontend

Your goal is to:

- Write a clean application that is solving the challenge
- Make sure the code is tested

Code Challenge: Customers are setting their advertising configuration in a dashboard. This dashboard is developed in php, using complex technologies and it is storing its data in several databases. However, every three minutes, a batch job is exporting dashboard configuration data into an XML files. The format of the xml is very simple and it looks like the following

```
<Creatives>
      <Creative id=”Video-1” price=”6.4567” currency=”EUR”/>
      <Creative id=”Video-4” price=”1.1234” currency=”USD”/>
      <Creative id=”Video-7” price=”55.123” currency=”SEK”/>
      <Creative id=”Video-12” price=”16.4567” currency=”EUR”/>
      <Creative id=”Video-25” price=”9.4567” currency=”USD”/>
</Creatives>
<Placements>
            <Placement id="plc-1" floor="1.3456" currency”EUR”/>
            <Placement id="plc-2" floor="90.234" currency=”SEK”/>
            <Placement id="plc-3" floor="8.343" currency=”TYR”/>
            <Placement id="plc-4" floor="20.56" currency=”USD”/>
            <Placement id="plc-5" floor="27.9856" currency=”EUR”/>
            <Placement id="plc-6" floor="22.5656" currency=”SEK”/>
            <Placement id="plc-7" floor="0" currency=”EUR”/>
            <Placement id="plc-8" floor="1.3456" currency=”USD”/>
</Placements>
```

Here you see that creative has three attributes:

- the id which is unique for each creative
- the price that indicates how worth is this creative
- The currency of the price

Placements are spaces in apps where creatives are going to be shown and they have three attributes:

- the id which is unique for each placement
- the floor price that indicates what is the minimal price of a creative in order to be shown
- The currency of the price

A creative with a price higher than a floor price of placement has the right to be shown on this placement, others cannot be shown. For example, **`Video-1`** can be shown on **`plc-2`** because **`6.4567>=4.234.`** On the other side, **`Video-1`** cannot be shown on **`plc-4`** because its price is too low.

The remaining part of the system is then accepting the configuration using google protobuf and the format of the message is the following

```
package FYBER.userconfiguration;
message Creative{
      required string id = 1;
      required float price = 2; //this is in EUR
};

message CreativeSeq{
      repeated Creative creative = 1;
};
message Placement{
      required string id = 1;
      repeated Creative creative = 2;
};

message PlacementSeq{
      repeated Placement placement = 1;
};
```

Your task is to create an application called AdParameters that reads the input xml file and creates a **`PlacementSeq`**. In order to show the output use a function that is printing the final protobuf to stdout. You see that in the message Placement there is a sequence of creatives; use the following rule in order to associate creatives to placement

- if price of a creative >= floor price of placement then the creative should be part of the sequence
- if price of a creative < floor price of placement then the creative should NOT be part of the sequence

Please take care of the following points:

- Make sure that you convert prices in one unique reference currency (EUR) and store everything in EUR inside the protobuf object
- Use the following exchange rate:
    - EUR TO TYR: 1:3.31
    - EUR TO USD: 1:1.13

BONUS: Create a web server that when requested a placement with a floor price returns the creative a creative that pays at least that floor price