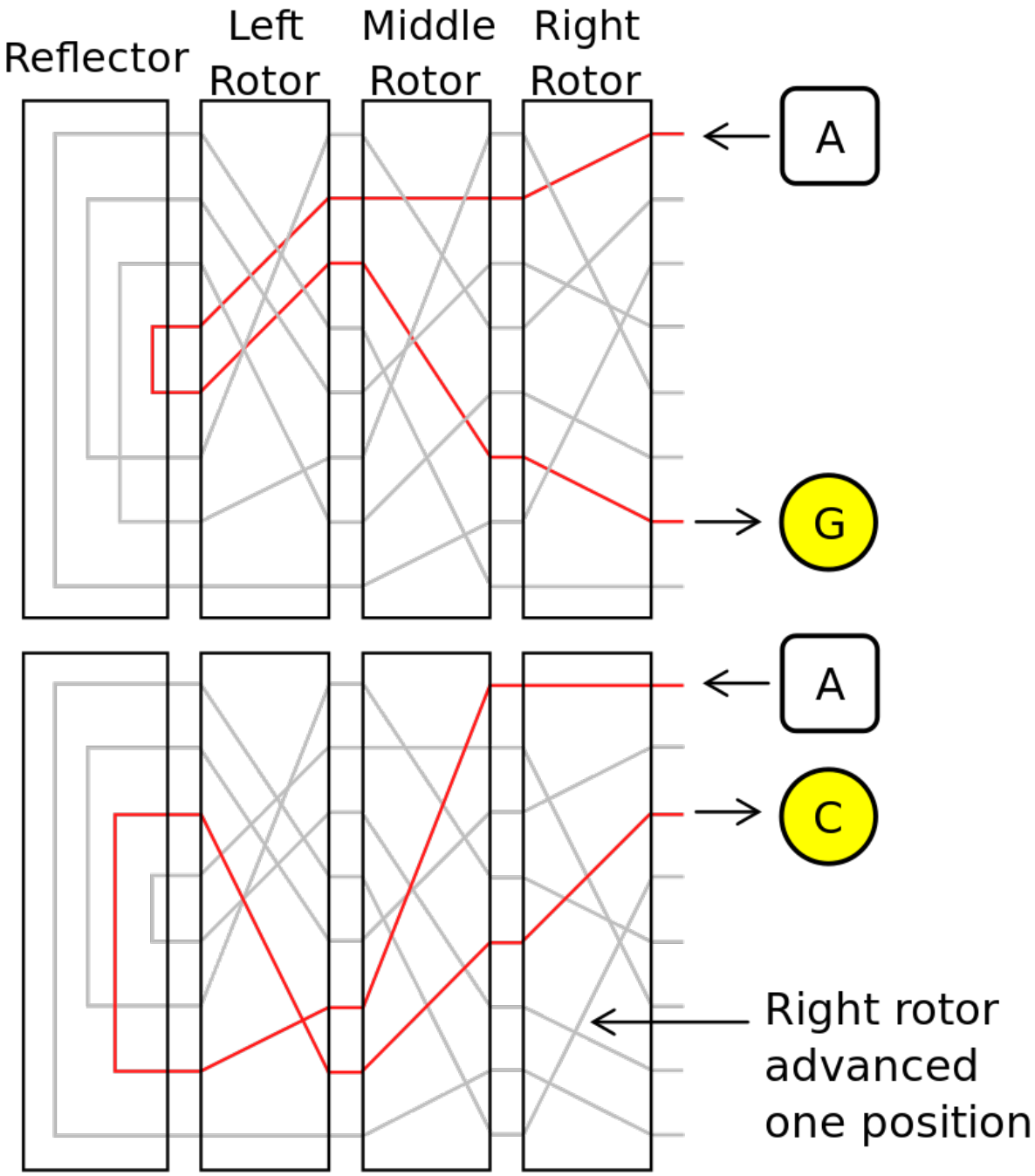
Exercise for Ovo

Good morning agent.
Your mission, should you choose to accept it, is to implement a simplified version of Enigma to encrypt the

message generated at the url: **http://www.randomtext.me/download/txt/gibberish/p-1/10-20**.

Enigma was an encryption machine used by the Germans during the second world war. Given a specific configuration it will encrypt a letter into another letter. A configuration is given by the positioning and inital position of 3 Rotors. A Rotor is a cylinder containing wires. For each letter, it will scramble the signal to another letter. For example for the rotor 1, A becomes B. And then the signals goes to the second rotor, then to the third. At the end there is a reflector that is in fact another rotor but feeds the signal back to the 3rd Rotor, then to the second and back to the first. The encrypted letter is then G.



Specs:

A simplified version of Enigma with 3 rotors only.

No plugboard
The rotors to use: Enigma I (https://en.wikipedia.org/wiki/Enigma_rotor_details)

```
input  ->|  |->|  |->|  |->|  |
output <-|  |<-|  |<-|  |<-|  |
```

Rotor I `[ekmflgdqvzntowyhxuspaibrcj]`
Rotor II `[ajdksiruxblhwtmcqgznpyfvoe]`
Rotor III `[bdfhjlcprtxvznyeiwgakmusqo]`
Reflector A `[ejmzalyxvbwfcrquontspikhgd]`

The rotors will turnover after a complete turn (no turnover or notches)

i.e. rotor x+1 will move one position after rotor x does a complete turn
The rotor moves when the key is pressed (before the signal goes through)

The input characters will be [a-z] only. Any uppercase should be transformed as lower case. The input is a random generated message at the URL:

http://www.randomtext.me/download/txt/gibberish/p-1/10-20

Punctuation and digits will be ignored. .

Expected Result:

Given **x** a randomly downloaded message from the given URL, **E(x)**

-> y with **y** the encrypted message.

Also **E(E(x)) -> x**: A encrypted message generated by an Enigma machine should be decypted by an identical Enigma machine given the same configuration.

The result can be send in an Android app with minimal UI

There is no time limit but 2-3 hours seems about the time required for the task (the solution is less that 200 lines of code)

If you can't manage to finish, write some pseudo code of what's left to do. This message will not self destruct :)

Contact

If something's not clear, do not hesitate to contact me at

idriss.juhoor@ovoenergy.com