



Test de compétence Nanocorp

C'est votre anniversaire. Un ami vous offre un puzzle un peu particulier, composé de câbles et de portes logiques. À vous de monter le circuit !

Détails du puzzle

Chaque câble a un identifiant (une suite de lettres minuscules) et peut transporter un signal sur 16 bits (un nombre de 0 à 65355). Un signal est transmis à un câble par une porte logique, un autre câble, ou une valeur spécifique directement. Un câble ne peut recevoir un signal que d'une seule source mais peut transmettre son signal à de multiples destinations. Une porte logique ne transmet son signal que si toutes ses entrées ont un signal.

Un livret d'instruction est fourni avec le puzzle et décrit comment connecter les différentes pièces : `x AND y -> z` vous indique que les câbles `x` et `y` sont connectés à une porte `AND` qui elle-même est connectée à au câble `z`.

Autres exemples :

- `1 -> x` indique que le signal `1` est transmis au câble `x`
- `x AND y -> z` indique que l'opération logique `ET` des câbles `x` et `y` est transmise au câble `z`
- `x LSHIFT 2 -> y` indique que le signal du câble `x` est décalé de `2` bits vers la gauche puis est transmis au câble `y`
- `NOT x -> y` indique que le signal du câble `x` est inversé et est transmis au câble `y`

Par ailleurs, sont aussi incluses dans la boîte du puzzle les portes logiques `OR` et `RSHIFT`.

Voici l'exemple d'un circuit simple :

```
123 -> x
456 -> y
x AND y -> d
x OR y -> e
```

```
x LSHIFT 2 -> f
y RSHIFT 2 -> g
NOT x -> h
NOT y -> i
```

Une fois le circuit construit, les câbles transportent les signaux suivants :

```
d: 72
e: 507
f: 492
g: 114
h: 65412
i: 65079
x: 123
y: 456
```

L'attendu

Sur la base des détails donnés ci-dessus, il vous est demandé de réaliser une **librairie** Rust dont l'API permettra de :

- construire un circuit à partir d'un fichier ou d'une chaîne de caractères représentant les instructions de branchement.
- pouvoir connecter un nouveau branchement au circuit à tout moment (que le circuit soit déjà construit ou non).
- pouvoir récupérer le signal d'un câble du circuit à tout moment.

Vous trouverez des données d'entrée dans le fichier `input.txt`.

Le rendu

Vous enverrez le résultat par mail contenant la librairie en entier (`Cargo.toml`, `src/`, ...) et les tests sous forme d'archive par mail au maximum 2 semaines après réception du test.

Vous serez jugé sur la qualité du code produit. Celui-ci devra être clair, documenté, testé, et future-proof (on doit pouvoir rajouter facilement des portes logiques par exemple). Ces sujets seront abordés lors de l'entretien présentiel.