

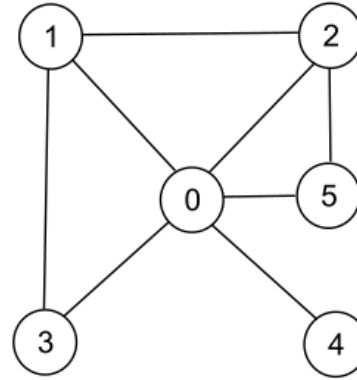
DS 3 - Terminale NSI (Correction)

Exercice 1

1.

```
voisins = [[1, 2, 3, 4],  
           [0, 2, 3],  
           [0, 1],  
           [0, 1],  
           [0]]
```

2.



3.

```
voisins = [[1, 2, 3, 4, 5],  
           [0, 2, 3],  
           [0, 1, 5],  
           [0, 1],  
           [0],  
           [0, 2]]
```

4.

```
def voisin_alea (voisins, s) :  
    return voisins[s][random.randrange(len(voisins[s]))]
```

5.

La fonction marche_alea s'appelle elle-même, c'est donc une fonction récursive.

6.

Cette fonction permet de parcourir une chaîne dans le graphe voisins, cette chaîne commence au sommet i et comporte n sommets. La fonction renvoie le dernier sommet de cette chaîne.

7.

```
def simule(voisins, i, n_tests, n_pas):  
    results = [0] * len(voisins)  
    for _ in range(n_tests):  
        j = marche_alea(voisins, i, n_pas)  
        results[j] += 1  
    return [total / n_tests for total in results]
```

8.

C'est l'ordinateur 0 car dans 32,8% des cas, le virus termine sa course sur l'ordinateur 0.

9.

```
def temps_propagation(voisins, s):  
    infectes = [0] * len(voisins)  
    infectes[s] = 1  
    pas = 0  
    tous_infectes = [1] * len(voisins)  
  
    print(infectes)  
    while infectes != tous_infectes:  
        nouveau_infectes = [0] * len(voisins)  
        for infecte in infectes:  
            sommet_aleatoire = marche_alea(voisins, infecte, 2) # Marche aléatoire  
            nouveau_infectes[sommet_aleatoire] = 1  
  
        for i in range(len(voisins)):  
            if nouveau_infectes[i]==1 and infectes[i]==0:  
                infectes[i]=1  
  
        print(infectes)  
        pas += 1  
  
    return pas
```

Exercice 2

Partie A

1. 255.255.0.0
2. 172.16.0.0
3. 172.16.255.255
4. $2^{16} - 2 = 65534$

Partie B

5. S1 -> A -> H -> D -> S2
6. S1 -> A -> H -> C -> D -> S2 ou S1 -> A -> B -> C -> D -> S2 ;
- 7.

Pour AHCD :

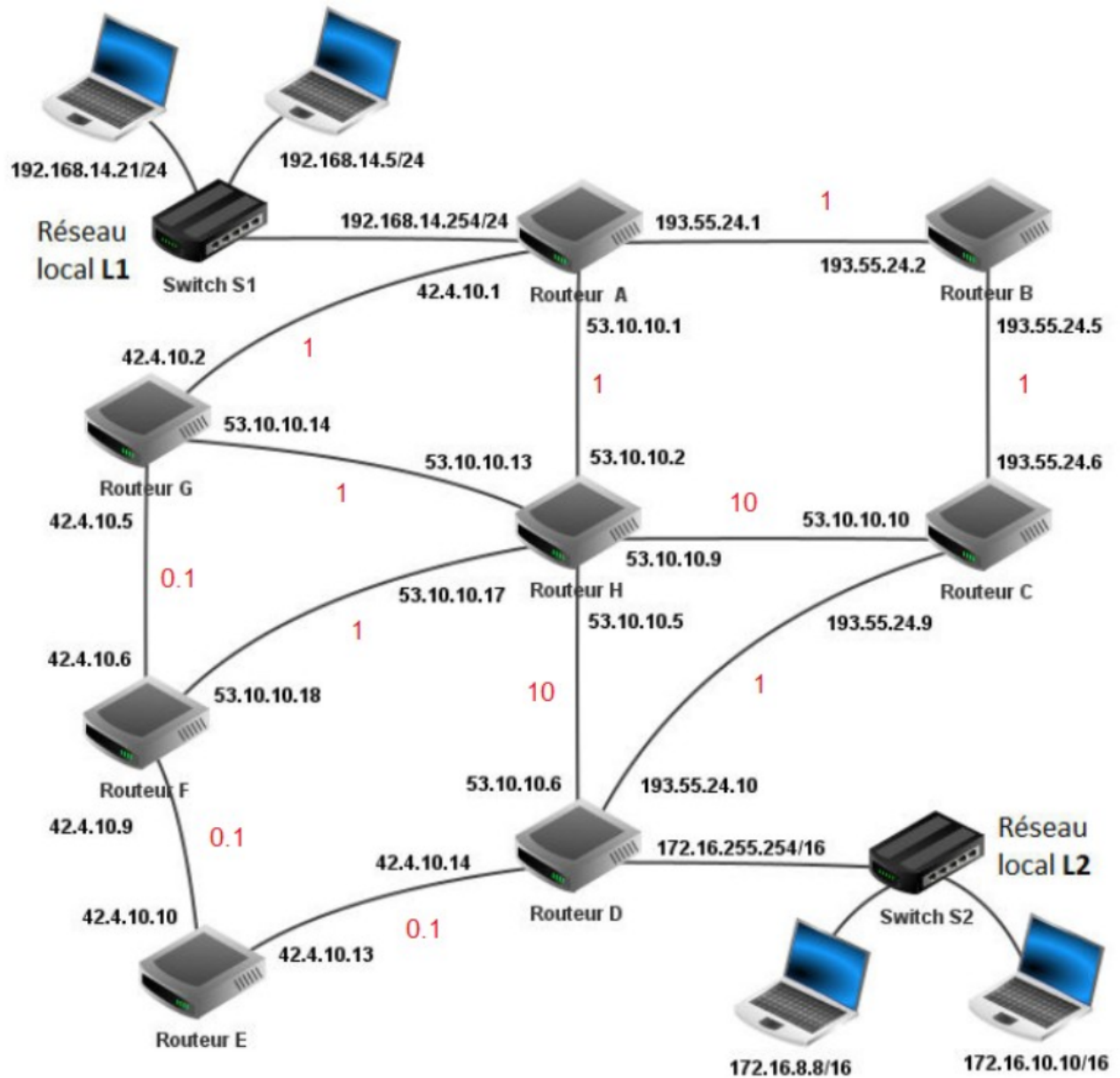
Routeur	Réseau destinataire	Passerelle	Interface
H	L2	53.10.10.10	53.10.10.9

Pour ABCD :

Routeur	Réseau destinataire	Passerelle	Interface
A	L2	193.55.24.2	193.55.24.1

- 8.
- 100 Mbit/s = 10^8 bit/s soit un coût = 10
 1 Gbit/s = 10^9 bit/s soit un coût = 1
 10 Gbit/s = 10^{10} bit/s soit un coût = 0,1

- 9.
- S1 -> A -> G -> F -> E -> D -> S2 avec un coût = 1 + 0,1 + 0,1 + 0,1 = 1,3



10. S1 -> A -> H -> F -> E -> D -> S2 avec un coût = 1 + 1 + 0,1 + 0,1 = 2,2

Exercise 3

1. Emplacement (id_emplacement:INT, nom:TEXT, localisation:TEXT, tarif_journalier: NUMERIC)

2.

1	myrtille	A4
4	mandarine	B1
6	melon	A2

- 3.
- ```
SELECT nom, prenom
FROM Client
WHERE ville = 'Strasbourg'
```