

EXERCICE 2 (3 points)

Cet exercice porte sur les réseaux.

Le réseau d'une entreprise dispose de quatre sites (SiteA, SiteB, SiteC et SiteD) et de cinq routeurs (R1, R2, R3, R4 et R5). La figure 1 en donne une représentation.

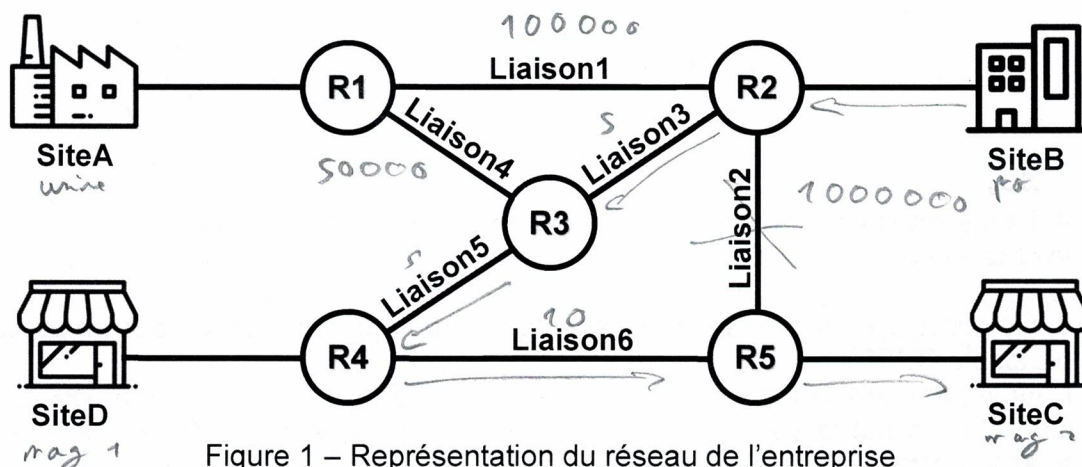


Figure 1 – Représentation du réseau de l'entreprise

1. Justifier brièvement que ce réseau sera encore fonctionnel même si l'une des liaisons (Liaison1, Liaison2, ..., Liaison6) est coupée.

car il y a toujours une autre liaison

2. Dans un premier temps, les tables de routage des routeurs sont configurées manuellement (voir figure 2).

Routeur R1		Routeur R2		Routeur R3	
Destination	Suivant	Destination	Suivant	Destination	Suivant
SiteA	Local	SiteA	R1	SiteA	R1
SiteB	R2	SiteB	Local	SiteB	R2
SiteC	R3	SiteC	R3	SiteC	R4
SiteD	R3	SiteD	R3	SiteD	R4

Routeur R4		Routeur R5	
Destination	Suivant	Destination	Suivant
SiteA	R3	SiteA	R4
SiteB	R3	SiteB	R4
SiteC	R5	SiteC	Local
SiteD	Local	SiteD	R4

Figure 2 – Tables de routage des routeurs R1, R2, R3, R4 et R5

Indiquer le chemin suivi par les paquets lorsqu'une information est envoyée de SiteB à SiteC.

3. Afin d'optimiser la maintenance du réseau, les tables de routage sont configurées automatiquement en utilisant le protocole RIP.

Pour le protocole RIP, le chemin est construit de façon à minimiser le nombre de routeurs traversés.

Recopier et compléter la table de routage RIP du routeur R1 (voir figure 3).

Routeur R1 (RIP)		
Destination	Suivant	Nombre de sauts
SiteA	Local	0
SiteB	R2	1
SiteC	R2	2
SiteD	R3	2

Figure 3 – Table de routage RIP du routeur R1

4. La liaison Liaison2 a un débit très inférieur aux autres liaisons. Expliquer pourquoi le choix du protocole RIP n'est pas judicieux. *PSK c'est lent*

5. On considère maintenant que les tables de routage sont configurées en utilisant le protocole OSPF.

Pour le protocole OSPF, le chemin est construit de façon à minimiser le coût. Le coût d'un chemin est la somme des coûts des liaisons à parcourir. Pour une liaison, la relation entre le coût (sans unité) et le débit D (en bit/s) est donnée par :

$$\text{coût} = \frac{10^{10}}{D}$$

Par convention, le coût d'une liaison directe entre un routeur et un site est 0. On donne le coût des liaisons dans la figure 4.

Liaison	Coût
Liaison1	100 000
Liaison2	1 000 000
Liaison3	5
Liaison4	50 000
Liaison5	5
Liaison6	10

Figure 4 – Table des coûts des liaisons

a. Indiquer la liaison dont le débit est le plus faible.

b. Donner la liste des quatre chemins possibles pour aller de SiteA à SiteC sans utiliser deux fois le même routeur et calculer le coût de chacun de ces chemins.

1,2 (1 100 000) < 1,3,5,6 (1 000 020) < 4,2,3 (1 050 005) < 6,5,6 (5 0015)

c. Recopier et compléter la table de routage OSPF du routeur R1 (voir figure 5).

Routeur R1 (OSPF)		
Destination	Suivant	Coût total du chemin
SiteA	Local	0
SiteB	R3	50005
SiteC	R3	50015
SiteD	R3	50005

Figure 5 – Table de routage OSPF du routeur R1

EXERCICE 3 (4 points)

Cet exercice porte sur les représentations binaires et les protocoles de routage.

1. Une adresse IPv4 est représentée sous la forme de 4 nombres séparés par des points. Chacun de ces 4 nombres peut être représenté sur un octet.
 - a. Donner en écriture décimale l'adresse IPv4 correspondant à l'écriture binaire :
11000000.10101000.10000000.10000011

192
168
128
131
 - b. Tous les ordinateurs du réseau A ont une adresse IPv4 de la forme :
192.168.128.___, où seul le dernier octet (représenté par __) diffère.
Donner le nombre d'adresses différentes possibles du réseau A. 256
2. On rappelle que le protocole RIP cherche à minimiser le nombre de routeurs traversés (qui correspond à la métrique). On donne les tables de routage d'un réseau informatique composé de 5 routeurs (appelés A, B, C, D et E), chacun associé directement à un réseau du même nom obtenues avec le protocole RIP :

Routeur A

Destination	Métrique
A	0
B	1
C	1
D	1
E	2

Routeur B

Destination	Métrique
A	1
B	0
C	2
D	1
E	2

Routeur C

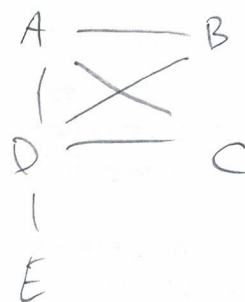
Destination	Métrique
A	1
B	2
C	0
D	1
E	2

Routeur D

Destination	Métrique
A	1
B	1
C	1
D	0
E	1

Routeur E

Destination	Métrique
A	2
B	2
C	2
D	1
E	0

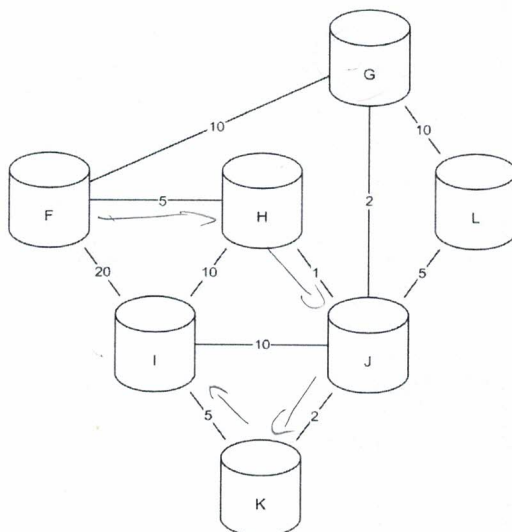


- a. Donner la liste des routeurs avec lesquels le routeur A est directement relié. *B, C, D*
- b. Représenter graphiquement et de manière sommaire les 5 routeurs ainsi que les liaisons existantes entre ceux-ci.
3. Le protocole OSPF est un protocole de routage qui cherche à minimiser la somme des métriques des liaisons entre routeurs.
 Dans le protocole de routage OSPF le débit des liaisons entre routeurs agit sur la métrique via la relation : $\text{métrique} = \frac{10^8}{\text{débit}}$ dans laquelle le débit est exprimé en bit par seconde (bps).
 On rappelle qu'un kbps est égal à 10^3 bps et qu'un Mbps est égal à 10^6 bps.
 Recopier sur votre copie et compléter le tableau suivant :

Débit	100 kbps	500 kbps	?	100 Mbps
Métrique associée	1 000	?	10	1

5000

4. Voici la représentation d'un réseau et la table de routage incomplète du routeur F obtenue avec le protocole OSPF :



Routeur F

Destination	Métrique
F	0
G	8
H	5
I	<i>43</i>
J	<i>6</i>
K	<i>8</i>
L	<i>11</i>

Les nombres présents sur les liaisons représentent les coûts des routes avec le protocole OSPF.

- a. Indiquer le chemin emprunté par un message d'un ordinateur du réseau F à destination d'un ordinateur du réseau I.
 Justifier votre réponse. *psk la liaison est plus rapide*
- b. Recopier et compléter la table de routage du routeur F.
- c. Citer une unique panne qui suffirait à ce que toutes les données des échanges de tout autre réseau à destination du réseau F transitent par le routeur G.
 Expliquer en détail votre réponse.

si h et k car la liaison avec J est deux fois plus lente

Exercice 3 (8 points)

Cet exercice porte sur l'architecture matérielle (réseau), les arbres binaires de recherche et la programmation Python.

L'entreprise CaféNet possède plusieurs cafés répartis dans différentes villes. Le réseau de la chaîne de cafés est représenté en Figure 1.

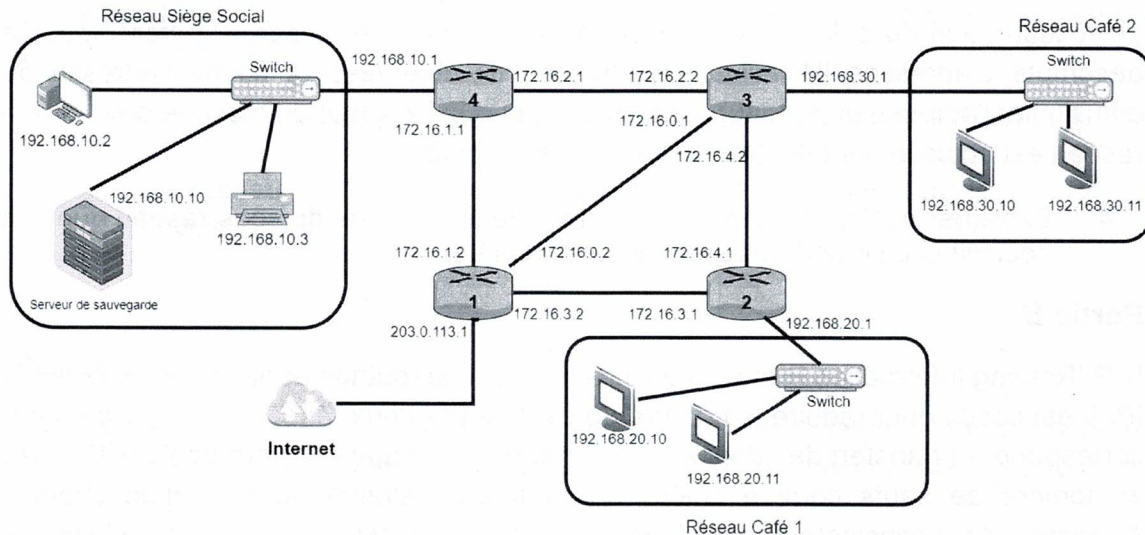


Figure 1. Schéma d'une partie du réseau

Sur le schéma sont représentés 4 routeurs, le réseau du siège social, le réseau du café 1, le réseau du café 2. Dans les réseaux du café 1 et du café 2, des bornes de commandes sont connectées à des switches (ce sont des boîtiers de connexion qui n'ont pas eux-mêmes d'adresse IP). Les 4 routeurs représentés sont composés d'au moins 3 interfaces réseau capable de relier des réseaux ensemble. Chaque interface possède donc une adresse IPV4 sur le réseau auquel elle est reliée.

Les masques des sous-réseaux sont tous 255.255.255.0. Avec ce masque, les trois premiers octets des adresses IP codent l'adresse réseau. Le dernier octet, c'est-à-dire les 8 derniers bits, code l'adresse des machines à l'intérieur de chaque sous-réseau.

Partie A

Le gérant veut faire installer une troisième borne de commande dans le café 1.

1. Indiquer les deux seules adresses IP valides pour cette nouvelle borne, parmi les quatre adresses IP proposées.

(a) 192.168.20.2

(b) 192.168.20.157

(c) 192.168.20.261

(d) 192.168.24.10

L'adresse de diffusion, appelée aussi adresse de broadcast, est la dernière adresse disponible à l'intérieur d'un réseau local. $192.168.20.255$

2. Déterminer l'adresse de diffusion du réseau du café 1.
3. Déterminer combien de machines informatiques il est encore possible de connecter au réseau du café 1 après l'installation de la troisième borne de commande. $254 - 3 = 251$

Le réseau local du café 1 n'a pas besoin de plus de 8 adresses IP différentes. Ce décompte d'adresses IP inclut les adresses IP réservées (à savoir l'adresse de diffusion et l'adresse du réseau). Il est rappelé que la longueur du masque de sous-réseau est actuellement de 24 bits (c'est-à-dire 3 octets).

4. Expliquer quelle est la longueur maximale du masque de sous-réseau que l'on pourrait choisir pour le réseau local du café 1.

Partie B

RIP (Routing Information Protocol) est un protocole de routage utilisé dans les réseaux IP. Il est conçu pour réduire le nombre de sauts entre deux réseaux. Un "saut" correspond au transfert des données d'un routeur à un autre. Le protocole RIP utilise le nombre de sauts comme critère principal pour évaluer le coût d'un chemin. Autrement dit, il considère que le chemin le plus optimal est celui qui traverse le moins de routeurs.

La table de routage du routeur 2 de la Figure 1 est représentée ci-dessous :

Routeur 2			
Réseau destination	Interface de sortie	Prochain routeur	Nombre de sauts
192.168.20.0	192.168.20.1	aucun	0
172.16.3.0	172.16.3.1	aucun	0
172.16.4.0	172.16.4.1	aucun	0
192.168.10.0	172.16.3.1	172.16.3.2	2
172.16.0.0	172.16.4.1	172.16.4.2	1
172.16.2.0	172.16.4.1	172.16.4.2	1
192.168.30.0	172.16.4.1	172.16.4.2	1
172.16.1.0	172.16.3.1	172.16.3.2	1

5. Recopier et compléter les deux dernières lignes de la table de routage du routeur 2.

La table de routage du routeur 2 contient un réseau de destination pour lequel deux routes différentes sont possibles. La ligne correspondante dans la table de routage aurait donc pu être remplie différemment tout en respectant le protocole RIP.

6. Identifier, dans la table de routage du routeur 2, le réseau de destination que l'on peut atteindre d'une autre façon et indiquer comment cette ligne de la table de routage pourrait être modifiée.

192.168.10.0 / 172.16.4.1 / 172.16.3.2

Une adresse IP qui n'est pas référencée dans la table de routage doit être routée par défaut vers Internet.

7. Recopier et compléter la ligne à ajouter à la table de routage du routeur 2.

Réseau destination	Interface de sortie	Prochain routeur
autre	<i>172.16.3.1</i>	<i>172.16.3.2</i>

Partie C

OSPF est également un protocole d'échanges de données entre les routeurs qui prend en compte le coût des routes. Le coût est lié au débit des liaisons entre les routeurs par la formule suivante :

$$\text{cout} = \frac{10^9}{\text{debit}} \text{ avec le débit en } \text{bit} \cdot \text{s}^{-1}.$$

8. Recopier et compléter la dernière colonne du tableau ci-dessous :

Tableau des coûts		
Type de connexion	Débit en $\text{bit} \cdot \text{s}^{-1}$	coût
Ethernet	$10 \text{ Mbit} \cdot \text{s}^{-1} = 10^7 \text{ bit} \cdot \text{s}^{-1}$	100
Fast Ethernet	$100 \text{ Mbit} \cdot \text{s}^{-1} = 10^8 \text{ bit} \cdot \text{s}^{-1}$	<i>10</i>
Fibre optique	$1 \text{ Gbit} \cdot \text{s}^{-1} = 10^9 \text{ bit} \cdot \text{s}^{-1}$	<i>1</i>

Le schéma ci-dessous met en évidence les types de connexion qui relient les routeurs.

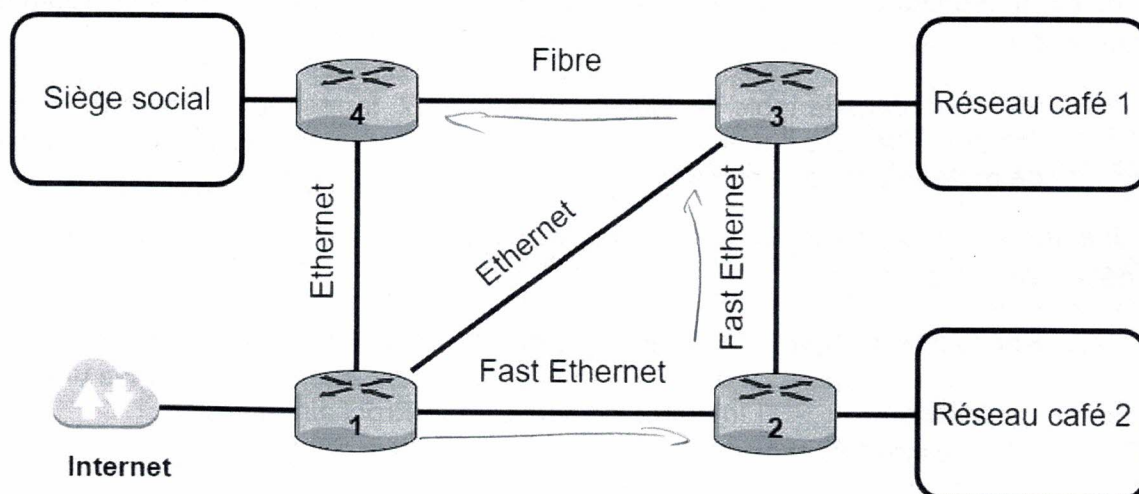


Figure 2. Schéma des types de connexion

9. Déterminer la route dont le coût est minimal pour aller du routeur 1 jusqu'au routeur 4 et calculer son coût au sens du protocole OSPF.

2 2

Partie D

Le but de cette partie est de classer les adresses IP des différents réseaux afin de faciliter leur recherche.

La fonction `ip_bin` prend en argument une chaîne de caractères décrivant une adresse IP en notation décimale, et renvoie une chaîne de caractères, de longueur 35 (32 bits et les 3 points), décrivant l'adresse IP en notation binaire.

Exemple :

```
>>> ip_bin('192.168.10.1')
'11000000.10101000.00001010.00000001'
```

10. Donner la chaîne de caractères renvoyée par `ip_bin('192.168.20.12')`.

11000000.10101000.00001010.00001100

La fonction `precede` prend en paramètres deux adresses IP en notation binaire, sous forme de chaînes de caractères identiques à celles renvoyées par la fonction `ip_bin`. La fonction `precede` renvoie un booléen qui vaut `True` si la première adresse IP en paramètre précède la seconde adresse IP.

Exemple :

```
>>> a = '11000000.10101000.00001010.00000001'
>>> b = '11000000.10101000.00001111.00000001'
>>> precede(a, b)
True
```

L'algorithme compare bit à bit les deux chaînes binaires, en lisant les chaînes de caractères dans le sens usuel (de gauche à droite). Dans l'exemple ci-dessus, tous les caractères sont identiques jusqu'au sixième caractère du troisième octet. Comme le bit de l'adresse a est inférieur à celui de l'adresse b, on en déduit que l'adresse IP a précède l'adresse IP b.

Si la première adresse IP ne précède pas la seconde, la fonction doit renvoyer `False`.

L'algorithme de comparaison est traduit dans le langage Python sous la forme suivante :

```
1 def precede(ip_1, ip_2):
2     for i in range(35):
3         if ip_1[i] < ip_2[i]:
4             return ...
5         elif ip_1[i] > ip_2[i]:
6             return ...
7     return ...
```

11. Expliquer dans quel cas la fonction `precede` exécutera la dernière instruction `return` de la ligne 7. *si $ip_1[i] = ip_2[i]$*

12. Recopier et compléter les lignes 4, 6 et 7 du code de la fonction `precede`.

Les tables de routage de chaque routeur sont implémentées sous la forme d'arbre binaire de recherche avec la classe `Abr`.

```
1 class Abr:
2     def __init__(self, adresse_ip,
3                 interface, passerelle,
4                 cout):
5         self.adresse_ip = adresse_ip
6         self.interface = interface
7         self.passerelle = passerelle
8         self.cout = cout
9         if adresse_ip != '':
10             self.gauche = Abr('', '', '', 0)
11             self.droite = Abr('', '', '', 0)
12
13     def est_vide(self):
14         return ...
```

Dans cette représentation :

- `adresse_ip` désigne l'adresse IP de la destination ;
- `interface` désigne l'interface réseau ;
- `passerelle` désigne l'adresse IP du prochain routeur ;
- `cout` désigne le nombre de sauts pour atteindre la destination.

- par convention, l'arbre binaire vide est une instance de `Abr` pour laquelle `adresse_ip` est une chaîne de caractères vide ;
 - un arbre binaire de recherche non vide possède nécessairement un sous-arbre gauche et un sous-arbre droit, éventuellement vides, qui sont tous les deux des arbres binaires de recherche. Ces sous-arbres sont désignés par `gauche` et `droite` dans la classe `Abr` ;
 - si elle n'est pas vide, l'adresse IP du sous-arbre gauche précède l'adresse IP de l'instance parent ;
 - si le sous-arbre droit n'est pas vide, alors l'adresse IP de l'instance parent précède l'adresse IP du sous-arbre droit.
13. Citer un attribut et citer une méthode de la classe `Abr`.
 14. Recopier et compléter la ligne 14 du code de la classe `Abr`.
 15. Justifier, en mobilisant des connaissances de cours, l'intérêt qu'il peut y avoir à représenter la table de routage par un arbre binaire de recherche.

La section de code qui définit `modifie` est incluse dans la classe `Abr`.

```

16     def modifie(self, adresse_ip,
17                 interface, passerelle,
18                 cout):
19         if self.est_vide():
20             self.adresse_ip = adresse_ip
21             self.interface = interface
22             self.passerelle = passerelle
23             self.cout = cout
24             self.gauche = Abr('', '', '', 0)
25             self.droite = Abr('', '', '', 0)
26         else:
27             self.adresse_ip = adresse_ip
28             self.interface = interface
29             self.passerelle = passerelle
30             self.cout = cout

```

Les lignes 20 à 23 sont exactement les mêmes que les lignes 27 à 30.

16. Réécrire le code de la fonction `modifie` en évitant cette répétition.

La classe `Abr` est complétée afin de permettre l'ajout de nouvelles lignes à la table de routage, tout en conservant les propriétés que doit posséder un arbre binaire de recherche.

```
32     def rechercher(self, adresse_ip):
33         if self.est_vide() or adresse_ip==self.adresse_ip:
34             return self
35         elif precede(...):
36             return self.gauche.rechercher(adresse_ip)
37         else:
38             return self.droite.rechercher(adresse_ip)
39
40     def inserer(self, adresse_ip,
41                interface, passerelle,
42                cout):
43         destination = self.rechercher(adresse_ip)
44         destination.modifie(adresse_ip,
45                             interface, passerelle,
46                             cout)
```

On rappelle que la fonction `precede` prend en arguments des adresses IP écrites sous forme binaire.

17. Recopier et compléter la ligne 35 du code de la fonction `rechercher`.