

Exercice 1

Partie A : L'adressage IP

1. (a) Le routeur F est sur le même réseau qu'un ordinateur d'adresse 192.168.5.3/24 (on a indiqué le masque réseau avec la notation CIDR), une adresse possible pour F est 192.168.5.1.
- (b) Avec une adresse réseau sur 24 bits et une adresse machine sur les 8 bits restants, il y a $2^8 = 256$ adresses possibles, mais il faut enlever l'adresse du réseau lui-même 192.168.5.0 et l'adresse de broadcast 192.168.5.255, ce qui laisse 254 adresses disponibles pour connecter des machines sur ce réseau (y compris la machine déjà connectée au switch R5 et y compris le serveur F).
2. (a) D'après la figure 1, le masque de sous-réseau de B est 255.255.240.0, ce qui s'écrit FF.FF.F0.00 en hexadécimal et 11111111.11111111.11110000.00000000 en binaire, ce qui signifie que l'adresse réseau est sur 24 bits soit /24 en notation CIDR.
- (b) En utilisant l'ordinateur d'adresse IP 192.168.2.2 et le masque 255.255.240.0, l'adresse réseau est obtenue par un AND et est donc 192.168.0.0/24.

adresse IP	a	1100 0000.	1010 1000.	0000 0010.	0000 0010
masque	b	1111 1111.	1111 1111.	1111 0000.	0000 0000
adresse réseau	a AND b	1100 0000.	1010 1000.	0000 0000.	0000 0000

- (c) L'intérêt d'avoir des liens redondants est essentiellement la tolérance aux pannes, le réseau reste connexe si un lien tombe en panne, et éventuellement la répartition de charge pour certains protocoles de routage adaptatifs qui sauront utiliser plusieurs chemins entre des réseaux (ce n'est le cas ni de RIP ni de OSPF).

Partie B : Le routage

1. (a) Avec RIP, il y a un seul chemin possible avec RIP, qui est A-B-E avec deux hops ; toujours avec RIP, il y a trois chemins possibles entre les routeurs F et B, de trois hops : F-H-G-B, F-H-E-B et F-D-A-B.
- (b) On donne ci-dessus les tables de routage de E et de G.

Destination	Routeur suivant	Distance
A	B	2
B	B	1
C	H	2
D	G	2
F	H	2
G	G	1
H	H	1

Destination	Routeur suivant	Distance
A	B, D	2
B	B	1
C	D, H	2
D	D	1
E	E	1
F	D, H	2
H	H	1

2. Avec le protocole OSPF les distances sont calculées avec les coûts $C = \frac{10^8}{D}$.

Destination	Routeur suivant	Coût total
A	D	1,1
B	H	10,11
C	D	1,1
D	D	0,1
E	H	10,1
G	D	1
H	H	0,1

3. Le chemin entre E et D sera E-G-D (de coût total 11).

Exercice 2

1. (a) La requête `SELECT AGE, TAILLE, POIDS FROM ANIMAL WHERE NOM='MORIS'` ; renvoie la table

age	taille	poids
6	1,70	100

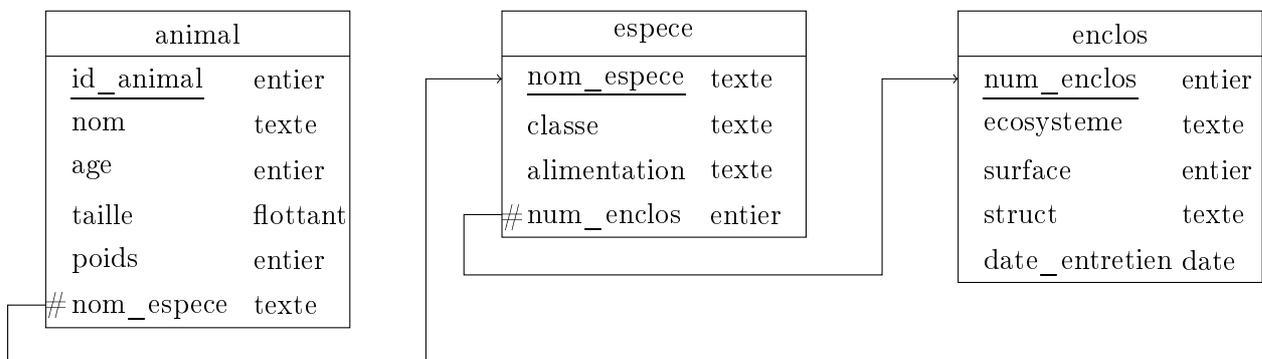
- (b) La requête `SELECT NOM, AGE FROM ANIMAL WHERE NOM_ESPECE='BONOBO' ORDER BY AGE` ; permet d'obtenir le nom et l'âge de tous les bonobos, triés par âge croissant.
2. La clé primaire de la table ESPECE est le champ `NOM_ESPECE` car cette clé est spécifique à chaque ligne, la clé étrangère est `NUM_ENCLOS` qui fait référence à la table ENCLOS.
3. On peut écrire le modèle relationnel ainsi :

ANIMAL(ID_ANIMAL : ENTIER, NOM : STRING, AGE : ENTIER, TAILLE : DECIMAL, POIDS : ENTIER, #NOM_ESPECE : STRING)

ENCLOS(NUM_ENCLOS : ENTIER, ECOSYSTEME : STRING, SURFACE : ENTIER, STRUCT : STRING, DATE_ENTRETIEN : DATE)

ESPECE(NOM_ESPECE : STRING, CLASSE : STRING, ALIMENTATION : STRING, #NUM_ENCLOS : ENTIER)

ou bien avec un schéma ; les noms des types dépendent du SGBDR.



4. (a) `UPDATE ESPECE SET CLASSE='MAMMIFÈRES' WHERE NOM_ESPECE='ORNYTHORYNQUE'` ; est une requête qui permet de corriger l'erreur mentionnée.
- (b) On insère le lama avec `INSERT INTO ANIMAL VALUES (179, 'SERGE', 0, 0.8, 30, 'LAMA')` ;
5. (a) Pour obtenir le nom et l'espèce de tous les animaux carnivores vivant en vivarium dans le zoo, on utilise la requête
- ```

SELECT NOM, NOM_ESPECE FROM ANIMAL
JOIN ESPECE ON ANIMAL.NOM_ESPECE = ESPECE.NOM_ESPECE
JOIN ENCLOS ON ESPECE.NUM_ENCLOS = ENCLOS.NUM_ENCLOS
WHERE ENCLOS.STRUCT='VIVARIUM' AND ESPECE.ALIMENTATION='CARNIVORE' ;

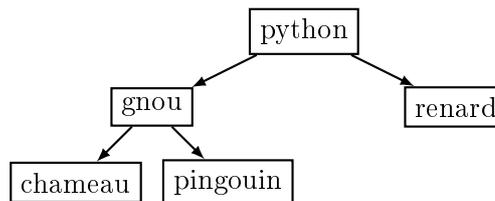
```
- (b) La requête `SELECT COUNT(*) FROM ANIMAL JOIN ESPECE ON ANIMAL.NOM_ESPECE = ESPECE.NOM_ESPECE WHERE ESPECE.CLASSE='OISEAUX'` ; permet de compter les oiseaux du zoo.

### Exercice 3

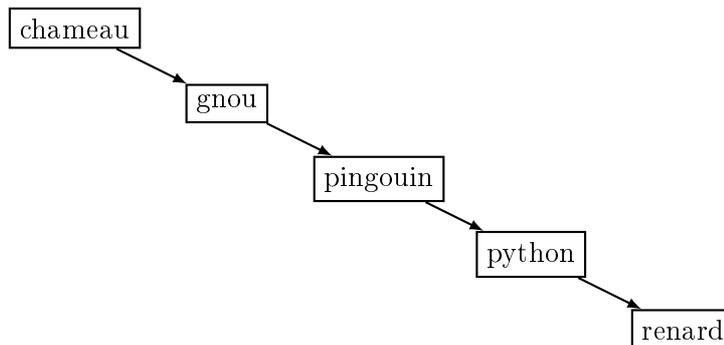
1. (a) L'exécution de `bonjour('Alan')` renvoie la chaîne `'Bonjour Alan !'`.
- (b) Les variables `x` et `y` contiennent des valeurs booléennes, respectivement `False` et `True`.
- (c) La fonction ci-dessous compte les occurrences de `une_lettre` dans `une_chaine`.

```
def occurrences_lettre(une_chaine, une_lettre):
 nb=0
 for caractere in une_chaine:
 if caractere == une_lettre:
 nb=nb+1
 return nb
```

2. (a) Avec 5 nœuds la hauteur minimale est 3 avec la définition imposée par l'énoncé (mais 2 avec la définition usuelle de la hauteur comme longueur du plus long chemin de la racine vers une feuille).



- (b) Et la hauteur maximale est 5, p.ex avec l'ABR ci-dessous.



3. (a) La valeur renvoyée par `mystere(abr_mots_francais)` est 336531 puisque dans la fonction `mystere` on reconnaît le calcul récursif de la taille d'un ABR.
- (b) On donne une fonction permettant de calculer la hauteur d'un ABR, ou plutôt son nombre de niveaux puisque c'est ce que demande l'énoncé.

```
def hauteur(un_abr):
 if un_abr.est_vide():
 return 0
 else:
 return 1+max(hauteur(un_abr.sous_arbre_gauche),
 hauteur(un_abr.sous_arbre_droit))
```

4. (a) On a complété la fonction `chercher_mots`.

```
def chercher_mots(liste_mots, longueur, lettre, position):
 res=[]
 for i in range(len(liste_mots)):
 if len(liste_mots[i])==longueur and liste_mots[i][position]==lettre:
 res.append(liste_mots[i])
 return res
```

- (b) La commande donne la liste des mots de longueur 3 qui contiennent un `'x'` en position 2 et un `'a'` en position 1.
- (c) Pour obtenir les mots de longueur 5 se terminant par `'ter'`, on peut utiliser `chercher_mots(chercher_mots(chercher_mots(liste_mots_francais,5,'t',2),5,'e',3),5,'r',4)`