

2022 Rattrapage jour 1

Exercice 1

Partie A

1. Arbre 1

Partie B

2.a. Le plus petit élément d'un arbre binaire de recherche est la feuille la plus à gauche car par construction, sa valeur est plus petite ou égale à toutes les autres valeurs.

2.b.

```
def RechercheValeur(ABR, cle):
    if est_vide(ABR):
        return False
    elif racine(ABR) == cle:
        return True
    else:
        return RechercheValeur(sous_arbre_gauche(ABR, cle) or
                               RechercheValeur(sous_arbre_droit(ABR, cle))
```

3.a. C'est le parcours infixe.

3.b. 7-2-1-5-3-6-10-8-9

3.c. 1-3-6-5-2-9-8-10-7

3.d. 7-2-10-1-5-8-3-6-9

Exercice 2

Partie A

1.a. 5

1.b. Les goélands

1.c.

```
def surface(self):
    return self.sejour.surface() + self.ch1.surface() + self.ch2.surface()
```

2.

```
for villa in v:
    if villa.equip() == "eq":
        print(villa.nom())
```

Partie B

3. Appel d'une fonction par elle-même.

4.

```
def max_surface(v):  
    if len(v) == 1:  
        return v[0]  
    else:  
        if v[0].surface() < v[1].surface():  
            v.pop(0)  
        else:  
            v.pop(1)  
        return max_surface(v)
```

Exercice 3

Partie A

1. Num_Objet est unique pour chaque objet, il peut donc être utilisé comme clé primaire.

2. Type (Type_Objet : String, Libelle_Objet : String)

Partie B

3. La b) car elle n'a pas de quotes autour des nombres et des quotes autour des chaînes de caractères.

4. Car la valeur 'BD' existe déjà pour la clé primaire Type_Objet.

5.

| | |
|-----------------|---------|
| Proxima Cen b | 768,067 |
| Lalande 21185 b | 392,753 |

6.

```
SELECT Nom_System, Nom_Objet, Libelle_Objet  
FROM Gaia  
JOIN Type ON Gaia.Type_Objet = Type.Type_Objet  
WHERE Parallaxe > 400 AND Libelle_Objet = 'Etoile';
```

7.a.

```
INSERT INTO Type VALUES('ST', 'Etoile');
```

7.b. On remplace toutes les références à '*' dans la table Gaia :

```
UPDATE Gaia  
SET Type_Objet = 'ST'  
WHERE Type_Objet = '*';
```

Puis on supprime la ligne '*' de la table Type :

```
DELETE FROM Type  
WHERE Type_Objet = '*';
```

Exercice 4

Partie A

1. a)

Partie B

2. 192.168.10.2

3. 254 si on enlève l'adresse du réseau et l'adresse de diffusion.

4. Un switch permet de relier les éléments d'un même sous-réseau. Il achemine les paquets entre les différents éléments.

5. Un routeur permet de connecter des sous-réseaux et d'acheminer des paquets en choisissant la meilleure route.

6.

| Destination Réseau | Passerelle | Métrieque |
|--------------------|------------|-----------|
| 192.168.10.0/24 | 0.0.0.0 | 0 |
| 2.100.40.0/24 | 2.100.40.1 | 1 |
| 3.100.30.0/24 | 3.100.30.2 | 1 |
| 4.10.10.0/24 | 4.10.10.2 | 1 |
| 4.20.10.0/24 | 4.10.10.2 | 2 |
| 7.30.40.0/24 | 3.100.30.2 | 3 |
| 6.10.30.0/24 | 4.10.10.2 | 3 |
| 90.10.20.0/24 | 2.100.40.1 | 2 |

7.

| Destination Réseau | Passerelle | Métrieque |
|--------------------|------------|-----------|
| 90.10.20.0/24 | 4.10.10.2 | 4 |

Exercice 5

1. Situation 2

2.a.

- V : Client3 | Client2 | Client1
- F : Client4
- val : Prioritaire

2.b.

```
def longueur_file(F):
    V = creer_file_vide()
    n = 0
    while not est_vide(F):
        n = n + 1
        val = defiler(F)
        enfiler(V, val)
    while not est_vide(V):
        val = defiler(V)
        enfiler(F, val)
    return n
```

2.c.

```
def compter_prio(F):
    V = creer_file_vide()
    n = 0
    while not est_vide(F):
        val = defiler(F)
        if val == 'Prioritaire':
            n = n + 1
        enfiler(V, val)
    while not est_vide(V):
        val = defiler(V)
        enfiler(F, val)
    return n
```