

# Proposition de correction

## Exercice 1

### Q1.a

jours[1]

### Q1.b

4

### Q2

```
numero_jour = (jours.index(element) + n ) % 7
```

### Q3.a

jour = mois[1][1]

### Q3.b

mois[(numero\_mois + 5) % 12][0]

### Q4.a

31

### Q4.b

```
def jour_suivant(date : tuple) -> tuple:
    nom_jour, numero_jour, numero_mois, annee = date
    numero_jour += 1
    if numero_jour > mois[numero_mois][1]:
        numero_mois += 1
        numero_jour = 1
    if numero_mois > 12:
        annee += 1
        numero_mois = 1
    return (jours[(jours.index(nom_jour) + 1) % 7], numero_jour, numero_mois, annee)
```

## Exercice 2

### Q1

```
panier1 = Panier()
panier1.enfiler((31002, "café noir", 1.50, 50525))
```

### Q2

```
def remplir(self, panier_temp):
    while not panier_temp.est_vide():
        article = panier_temp.defiler()
        self.enfiler(article)
```

### Q3

```
def prix_total(self) -> float:
    panier_temp = Panier()

    total = 0
    while not self.est_vide():
        article = self.defiler()
        panier_temp.enfiler(article)
        total += article[2]

    #on reconstruit le panier original
    self.remplir( panier_temp )

    return total
```

plus simple mais hors propos...

```
def prix_total(self) -> float:
    total = 0
    for article in self.file:
        total += article[2]
    return total
```

### Q4

```
def duree_courses(self) -> int:
    premier_article, dernier_article = 0, 0
    if not self.est_vide():
        premier_article = self.defiler()[3]
    while not self.est_vide():
        dernier_article = self.defiler()[3]
    #on ne peut donner la durée des achats pour 1 seul article
```

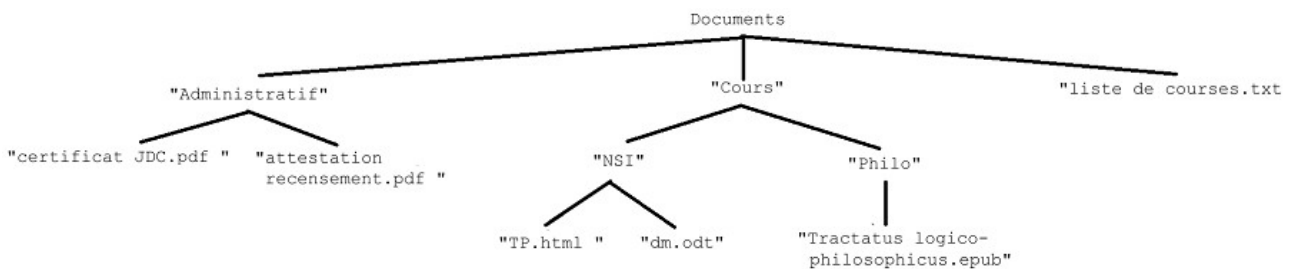
```
#le temps renvoyé sera < 0
return dernier_article - premier_article
```

plus simple mais hors propos...

```
def duree_courses(self) -> float:
    duree = 0
    if len(self.file) > 1:
        duree = self.file[len(file) - 1][2] - self.file[0][2]
    return duree
```

### Exercice 3

#### Q1



#### Q2.a

```
def Parcourir(racine, adr):
    dossier = racine
    for nom_dossier in adr:
        dossier = dossier[nom_dossier]
    return dossier
```

#### Q2.b

60

#### Q3.a

```
def Ajouter_fichier(racine, adr, nom_fichier, taille):
    dossier = Parcourir(racine, adr)
    dossier[nom_fichier] = taille
```

#### Q3.b

```
def Ajouter_dossier(racine, adr, nom_dossier)
    dossier = Parcourir(racine, adr)
    dossier[nom_dossier] = {}
```

## Exercice 4

### Q1.a

id\_mesure : la valeur est unique

### Q1.b

id\_centre

### Q2.a

Affiche tous les attributs (id\_centre, nom\_ville, latitude, longitude et altitude) des Centres météorologiques dont l'altitude est > à 500 m

138	Grenoble	45.185	5.723	550
185	Tignes	45.469	6.909	2594
126	Le Puy-en-Velay	45.042	3.888	744
317	Gérardmer	48.073	6.879	855

### Q2.b

```
SELECT nom_ville
FROM Centres
WHERE altitude BETWEEN 700 AND 1200
ORDER BY nom_ville
```

### Q2.c

```
SELECT longitude, nom_ville
FROM Centres
WHERE longitude > 5
ORDER BY nom_ville
```

### Q3.a

Affiche toutes les mesures et leurs les attributs (id\_mesure, id\_centre date, temperature, pression et pluviometrie) prises le 30 octobre 2021

2174	126	2021-10-30	18.2	1023	0
2200	185	2021-10-30	5.6	989	20

### Q3.b

```
INSERT INTO Mesures
VALUES(3650, 138, '2021-11-8', 10, 1013, 0)
```

**Q4.a**

Affiche tous les attributs (id\_centre, nom\_ville, latitude, longitude et altitude) des Centres météorologiques situés les plus au sud (ici Nice)

**Q4.b**

```
SELECT DISTINCT Centres.nom_ville
FROM Centres, Mesures
WHERE Mesures.temperature < 10 AND Mesures.date BETWEEN 2021-10-1 AND 2021-10-31
AND Centres.id_centre = Mesures.id_centre
ORDER BY Centres.nom_ville
```

**Exercice 5**

---

**Q1.a**

Le SoC intègre la mémoire et les périphériques.

**Q1.b**

Le Broadcom BCM271 intègre une interface réseau

**Q1.c**

- L'architecture 32/64 bits
- La fréquence

**Q2.a**

Adresse physique

- les 3 premiers chiffres en hexa indique le constructeur
- Les 3 autres suivants le n° de série de la carte

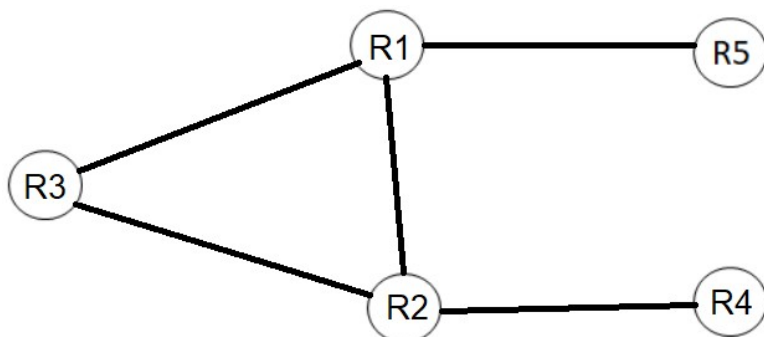
**Q2.b**

L'adresse IP de l'interface réseau (adresse logique)

**Q2.c**

Adresse de la passerelle (routeur)

**Q3.a**



**Q3.b**

$R4 \rightarrow R2 \rightarrow R1 \rightarrow R5$

**Q4**

$R4 \rightarrow R2 \rightarrow R3 \rightarrow R1 \rightarrow R5$  (coût = 22)