

Éléments de correction sujet 14 (2024)

Exercice 1

1. $i : 7, 8, 9, 10$; $f1(7)$ se termine
2. $f1(-2)$ se termine, la valeur renvoyée est 10
3. $i : 12, 13, 14, 15, 16\dots$; ne se termine pas (car toujours différent de 10)
4. $f1(n)$ se termine pour tous les entiers strictement inférieurs à 11.
5. $f2(4)$ se termine et renvoie 6
6. $f2(5)$ ne se termine pas, car nous avons les appels récursifs suivants : $f2(3)$, $f2(1)$, $f2(-1)$, $f2(-3)\dots$ on ne rencontre jamais le cas de base ($n=0$)
7. Tous les entiers pairs supérieurs ou égaux à 0.
8.

```
def infini(n):  
    return n + infini(n-1)
```
9. on appelle $\text{infini}(42)$ et donc $\text{paradoxe}(\text{code_paradoxe})$ ne termine pas.
10. on exécute `return 0`, donc $\text{paradoxe}(\text{code_paradoxe})$ termine.
11. Il y a un paradoxe (la fonction arrêt se termine quand elle ne termine pas), la fonction arrêt ne peut donc pas exister.

Exercice 2

1.

étape 1

3
2
1

moitié1

4
5
6

jeu

moitié2

étape 2

3
2
1

moitié1

jeu

6
5
4

moitié2

étape 3

4
1
5
2
6
3

moitié1

jeu

moitié2

2.

```
def produire_jeu(n):
    resultat = Pile()
    for i in range(n):
        resultat.empile(n - i)
    return resultat
```
3.

```
ligne 3 : Pile()
ligne 4 : for i in range(n//2)
ligne 6 : for i in range(n//2)
```
4.

```
def recombinaer(m1, m2):
    m = Pile()
    while not m1.est_vide():
        m.empile(m1.depile())
        m.empile(m2.depile())
    return m
```
5.

```
def faro(p, n):
    p1, p2 = scinder_jeu(p, n)
    return recombinaer(p1, p2)
```
6.

```
p1 = Pile()
p1.empile(1)
p2 = Pile()
assert not identiques(p1, p2)
p3 = Pile()
p3.empile(5)
p4 = Pile()
p4.empile(8)
assert not identiques(p3, p4)
p5 = Pile()
p5.empile(5)
p6 = Pile()
p6.empile(5)
assert identiques(p5, p6)
```
7.

```
def ordre_faro(n):
    compteur = 1
    p_init = produire_jeu(n)
    m = faro(produire_jeu(n), n)
    while not identiques(p_init, m):
        compteur = compteur + 1
        m = faro(m, n)
    return compteur
```

Exercice 3

1. 4 octets
2. Serveur_web : 172.16.0.1 ; Serveur_BDD : 172.16.0.2
3.
Permet de tester la connexion entre 2 machines.
4.
La passerelle par défaut est mal configurée, il faut indiquer l'adresse IP suivante :
192.168.1.254 (192.168.0.254 ne correspond pas au réseau local de PC_A1)
5.
PC_A1 -> Routeur A -> Routeur B -> Routeur C -> Routeur D -> Serveur_impression
6.
PC_A1 -> Routeur A -> Routeur B -> Routeur C -> X ; les paquets n'arrivent pas à destination
- 7.

Routeur C		
Destination	Prochain saut	Métrieque
172.16.0.0	10.0.2.2	2
192.168.0.0	10.0.2.2	2
192.168.1.0	10.0.2.2	2
192.168.2.0	10.0.3.2	1
192.168.3.0	10.0.4.2	1
10.0.0.0	10.0.2.2	1
10.0.1.0	10.0.2.2	1
10.0.2.0	-	-
10.0.3.0	-	-
10.0.4.0	-	-
10.0.5.0	10.0.3.2	1
0.0.0.0	10.0.2.2	2

8.
PC_A1 -> Routeur A -> Routeur B -> Routeur C -> Routeur D -> Serveur_impression
9.
Le débit de la liaison entre C et D est faible (10 Mb/s), il vaut mieux traverser un routeur de plus en passant par le routeur E car les débits sont meilleurs (1Gb/s)

10.

192.168.2.0	10.0.4.2	2
10.0.5.0	10.0.4.2	1

PC_A1 -> Routeur A -> Routeur B -> Routeur C -> Routeur E -> Routeur D ->
Serveur_impression

11.

```
SELECT titre_parution  
FROM parution
```

12.

Permet de sélectionner le numéro de parution et le numéro des pages qui ont pour police de caractère Arial,12. Ces informations seront classées par ordre de numéro de parution

13.

```
SELECT num_image, titre_image, poids  
FROM image  
WHERE poids > 1000
```

14.

Permet d'obtenir les numéros des parutions qui contiennent des images avec le terme Appolo dans leur titre.

15.

On ajoute une image (numéro 2923) à la table image. Cette image a pour titre Volcans du massif central, une largeur et une hauteur de 400 pixels et un poids de 1430 Ko.

16.

```
INSERT INTO texte  
VALUES  
(2754, 'Vulcania', 'Parc d'attraction', 250)
```

17.

On supprime de la table texte le texte ayant pour numéro 2034

18.

```
DELETE FROM comporte_texte  
WHERE num_texte = 2034
```