

Exercices divers



1. Complexité et coût

Exercice 1

Soit un algorithme de complexité (coût d'exécution dans le pire cas en terme de nombre d'opération) donné par $f(n)$ où n est la taille des données en entrée.

Associer à chaque fonction sa description.

(a) $f(n) = 2^n$ (b) $f(n) = \log_2(n)$ (c) $f(n) = n \log_2(n)$ (d) $f(n) = n$ (e) $f(n) = n^2$

1. le coût est prohibitif sauf pour de (très) petites entrées (comme $n = 10$)
2. le coût augmente très lentement lorsque la taille de l'entrée augmente
3. le coût augmente de façon proportionnelle à la taille de l'entrée
4. le coût augmente de façon un peu plus rapide qu'une fonction linéaire
5. le coût devient prohibitif pour de grandes entrées (comme $n = 10^6$)

Exercice 2

Soit t un tableau comportant n éléments. Donner le meilleur coût, avec un algorithme efficace, pour trouver dans t un élément e dans chacun des cas :

1. le tableau t est trié.
2. le tableau t n'est pas trié.

Indiquer dans chaque cas la méthode utilisée.

Exercice 3

Combien d'opérations au pire nécessitent ces opérations suivantes lorsqu'elles effectuées de manière optimales ? Dans chaque cas, la réponse est soit une seule, soit le nombre d'éléments de l'entrée.

1. Rechercher si un élément existe dans une file, une pile ou une liste.
2. Rechercher si un élément existe dans les clés d'un dictionnaire.
3. Rechercher si un élément existe dans les valeurs d'un dictionnaire.

2. Types

Exercice 4

Quelles affirmations sont correctes ?

1. Un élément d'un tuple peut être un tuple.
2. On peut ajouter un élément à un tuple.
3. Les éléments d'un tableau sont immuables.
4. Les éléments d'un tableau peuvent être des tableaux.
5. Les clés d'un dictionnaire peuvent être muables.
6. Les valeurs d'un dictionnaire peuvent être muables.

Exercice 5

On considère le code suivant :

```
for e in x:  
    print(e)
```

Quels types pour `x` sont possibles parmi les suivants ?

Dans ces cas possibles, expliquer ce que sont les `e` qui sont affichés à l'exécution du code.

1. une chaîne de caractère
2. une liste
3. un dictionnaire
4. un tuple

Exercice 6

Quel programme affiche le contenu du tableau `t` en ordre inverse sans erreur ?

1.

```
for i in range(len(t),0,-1):  
    print(t[i])
```

3.

```
for i in range(1, len(t)+1):  
    print(t[len(t)-i])
```

2.

```
for i in range(len(t),0,-1):  
    print(t[i-1])
```

4.

```
for i in range(len(t)):  
    print(t[len(t)-i])
```

Exercice 7

On exécute les instructions suivantes :

```
z = 'z'  
d = {'x': 10, 'y': 20, z: 30}
```

Quelles sont les instructions valides parmi les suivantes ?

1. `d[0]` 2. `d[10]` 3. `d[x]` 4. `d[z]` 5. `d['x']` 6. `d['z']`

Exercice 8

Quelle est la valeur affichée par ce programme ?

```
t = [10, 20, 30]  
s = t  
t[1] = 2  
print(s[1])
```

Exercice 9

Quelles sont les valeurs affichées par ce programme ?

```
def mystere(p):  
    p['x'] = 100  
    return p  
p = {'x': 10, 'y': 20}  
p2 = mystere(p)  
print(p['x'], p2['x'])
```

3. Classes et objets

Exercice 10

En Python, qu'est-ce que le constructeur d'une classe ?

Exercice 11

Soit f une méthode sans argument de la classe C et o un objet de C .

Quelle est la syntaxe pour exécuter la méthode f de o ?

Exercice 12

On considère une classe `MaClasse` ayant un attribut `age`.

Quelle est la valeur de `a.age` après exécution du code suivant ?

```
a = MaClasse()  
a.age = 15  
b = a  
b.age = 10
```