

Test de fin de semestre: Solutions

QCM sur la partie théorique.

Rappel: Pour $a, b \in \mathbb{R}$, $\cos(a - b) - \cos(a + b) = 2 \sin(a) \sin(b)$.

QCM sur la partie programmation.

Question 11. Si \mathbf{s} est une chaîne de p caractères, combien de caractères comporte la chaîne retournée par l’expression $\mathbf{s}[\mathbf{n:m}]$, avec $n \leq m \leq p$?

- (A) $m - n$
- (B) $m - n + 1$
- (C) $p - (m - n)$
- (D) $m - n - 1$

Question 12. Qu’affiche ce code, si les deux appels `cond1()` et `cond2()` renvoient `False`?

```
if not cond1():  
    print("A")  
if cond2():  
    print("B")  
else:  
    print("C")
```

- (A) **A**
 B
- (B) **A**
- (C) **c**
- (D) **A**
 C

Question 13. Qu'affiche ce code sur le terminal?

```
values = [True, False, True, False, True, False]
i = 0
while values[i]:
    print(i)
    i += 1
```

(A) Rien

(B) 0

(C) 0

2

4

(D) 1

3

5

Question 14. Qu'affiche ce code sur le terminal?

```
inds = [3, 0, 2, 1]
for i in inds:
    print(inds[i])
```

(A) 1

3

2

0

(B) 3

0

2

1

(C) 0

1

2

3

(D) Une erreur

Question 15. Imaginons une fonction **f** définie comme ci-dessous. Quel appel de cette fonction n'est *pas* possible?

```
def f(a: int, s: Optional[str], b: float = 0.1) -> None
```

(A) **f(42)**

(B) **f(s="42")**

(C) **f(b=42.0, a=42)**

(D) Aucun de ces trois appels n'est possible.

Question 16. Quelle affirmation sur les fonctions est *fausse* ?

- (A) Si une fonction a un autre type de retour que **None**, elle doit livrer une valeur de retour de ce type avec un **return**.
- (B) Si une fonction a **None** comme type de retour, elle ne peut pas contenir le mot clé **return**.
- (C) Une fonction peut contenir le mot clé **return** plusieurs fois.
- (D) Une instruction suivant immédiatement un **return** ne sera pas exécutée vu que le **return** indique la fin de l'exécution du code de la fonction.

Question 17. Quelle expression sous forme de compréhension de liste ci-dessous ne génère *pas* la liste des premières puissances de 2 ?

- (A) `[2 ** i for i in range(5)]`
- (B) `[n ** i for n in [2] for i in range(0, 5, 1)]`
- (C) `[2 ** n for n in [1 for i in range(5)]]`
- (D) `[2 ** i for i in [0, 1, 2, 3, 4]]`

Question 18. Si **data** est une liste de **floats**, quelle ligne affichera sur le terminal systématiquement *tous* les éléments de cette liste ?

- (A) `print(data[:])`
- (B) `print(data)`
- (C) `print(data[0:])`
- (D) Plusieurs solutions ci-dessus sont correctes.

Question 19. Si **f** est une fonction qui accepte un **int** et qui retourne une liste de **int**, quel est son type ?

- (A) `Callable[int, List[int]]`
- (B) `Callable[[int], List[int]]`
- (C) `Callable[[int, int], List]`
- (D) `Callable[List, int]`

Question 20. Quelle affirmation sur les threads est incorrecte ?

- (A) Un thread peut être créé par un autre thread.
- (B) Dans une application à interface graphique (comme avec Tkinter), on doit créer un nouveau thread pour gérer chaque événement (clic de souris, frappe clavier, redimensionnement de la fenêtre, etc.).
- (C) Les threads permettent d'exécuter plusieurs séquences d'instructions de manière concurrente (voire vraiment parallèle si la machine le permet).
- (D) C'est le système d'exploitation qui décide de quand un thread pourra exécuter son code et de quand il sera obligé de faire une pause.