

Correction : Évaluation sur les Types de Données Abstraites et Programmation en Python

Partie I : QCM et Questions Théoriques

Section A : QCM (60 points)

1. Un ordinateur peut démarrer sans BIOS
Réponse : **b. Faux**
2. Qu'est-ce qu'un système d'exploitation ?
Réponse : **b. Un logiciel permettant de gérer les ressources matérielles et logicielles d'un ordinateur.**
3. Quel composant est le "cerveau" de l'ordinateur ?
Réponse : **c. Le processeur (CPU).**
4. Quelle est la différence entre une liste et un dictionnaire en Python ?
Réponse : **b. Une liste est ordonnée ; un dictionnaire utilise des clés pour accéder aux éléments.**
5. Qu'est-ce qu'un algorithme ?
Réponse : **b. Une séquence d'instructions permettant de résoudre un problème.**
6. Quelle méthode permet d'ajouter un élément à une liste en Python ?
Réponse : **c. append()**
7. Que fait l'instruction break dans une boucle ?
Réponse : **a. Elle interrompt la boucle immédiatement.**
8. Quel type de donnée Python retourne la fonction input() ?
Réponse : **b. Str**
9. Quelle est la syntaxe correcte pour définir une fonction en Python ?
Réponse : **c. def nom_de_fonction():**
10. Que retourne la fonction len() appliquée à une liste ?
Réponse : **b. Le nombre d'éléments dans la liste.**
11. Comment accède-t-on au troisième élément d'une liste L en Python ?
Réponse : **c. L[2]**
12. Il est possible de démarrer un ordinateur sans carte mère.
Réponse : **b. Faux**
13. Qu'est-ce qu'un TDA ?
Réponse : **b. Une structure de données définie par son comportement et non son implémentation.**

14. Parmi les structures suivantes, laquelle est un TDA ?

Réponse : **b. Liste chaînée**

15. Quel type d'accès est permis dans une pile ?

Réponse : **b. LIFO (Last In, First Out).**

16. Quel TDA est utilisé pour gérer une file d'attente ?

Réponse : **b. File**

17. Quelle est la complexité moyenne de la recherche dans un tableau non trié ?

Réponse : **b. $O(n)$**

18. Quelle méthode permet d'ajouter un élément à la fin d'une liste en Python ?

Réponse : **b. `append()`**

19. Quelle est la principale différence entre un tableau statique et une liste en Python ?

Réponse : **b. Le tableau a une taille fixe, tandis que la liste peut être modifiée dynamiquement.**

20. Que fait la méthode `pop()` d'une pile ?

Réponse : **b. Supprime et retourne l'élément au sommet de la pile.**

Section B : Questions ouvertes (20 points)

1. Architecture d'un ordinateur (5 points)

- **Processeur (CPU)** : Effectue les calculs et exécute les instructions des programmes.
- **Mémoire vive (RAM)** : Stocke temporairement les données et instructions utilisées par le CPU.
- **Disque dur (HDD/SSD)** : Permet de stocker les données de manière permanente.

2. Langages de programmation (5 points)

- **Langage interprété** : Traduit et exécute le code ligne par ligne (exemple : Python).
- **Langage compilé** : Traduit tout le code source en un fichier exécutable avant l'exécution (exemple : C++).

3. Structures de contrôle (5 points)

- **Boucle for** : Utilisée pour parcourir un intervalle ou un ensemble fini d'éléments.

Exemple :

```
for i in range(5):  
    print(i)
```

- **Boucle while** : Utilisée pour répéter tant qu'une condition est vraie.

Exemple :

```
i = 0  
while i < 5:  
    print(i)  
    i += 1
```

4. Fonctions (5 points)

- Importance : Les fonctions permettent de structurer un programme, de réutiliser du code et de simplifier le débogage.
- Concepts :
 - **Paramètres** : Données passées à la fonction.
 - **Valeurs de retour** : Résultats renvoyés par la fonction.
 - **Portée des variables** : Les variables locales existent uniquement dans la fonction, tandis que les variables globales sont accessibles partout.

Partie II : Exercices Pratiques (40 points)

1. Types simples et conditions (20 points)

```
temperature = float(input("Entrez une température en Celsius : "))
if temperature <= 0:
    print("L'eau est à l'état solide.")
elif 0 < temperature < 100:
    print("L'eau est à l'état liquide.")
else:
    print("L'eau est à l'état gazeux.")
```

2. Structures conditionnelles avancées (20 points)

```
def determiner_categorie(age):
    if age < 12:
        return "Enfant"
    elif 12 <= age <= 17:
        return "Adolescent"
    elif 18 <= age <= 64:
        return "Adulte"
    else:
        return "Senior"
```