

Programmation – Algorithmique L1 – S1

Contrôle Continu 1 – jeudi 13/11/2025

Durée : 1h 30 minutes (2h Tiers-Temps)

Sujet 1

Nom :

Prénom :

Numéro d'étudiant :

- L'usage de la calculatrice, de tout appareil électronique (dont smartphones et montres connectées) est interdit et sera assimilé à un cas de triche.
- Aucun document autorisé.
- Dans la partie QCM : 1 seule réponse correcte par question. Aucune réponse : 0 pt, réponse fausse : -0,25 pt, réponse correcte : les points de la question.
- Si vous répondez « une autre réponse », vous devez indiquer votre réponse

Partie QCM (5 points)

Question 1 (1 point)

Donnez la valeur de `15 - 3 * 7 // 4 % 5 + 3 + 2 * 4 - 1 + 43 % 11 // 7`

☐ 12

☐ 26

☐ une autre réponse :

Question 2 (1 point)

Donnez la valeur de `(42 // 4) / (42 % 4)`

☐ 5

☐ 5.25

☐ une autre réponse :

Question 3 (1 point)

que vaut `not(True and not True) or 27%7 != 3*2 and not (3 != 7//2)`

☐ True

☐ False

Question 4 (1 point)

Donnez la valeur de `'4' < '17'`

☐ True

☐ False

Question 5 (1 point)

Que donne ce code ?

```
1 ch = "algorithmesDeTriBasiques"
2 print(ch[4:15:3] + ch[-11:-7])
```

☐ "rhsriBa"

☐ "rhsTBTriB"

☐ "rhsTTriBa"

☐ "rhsTTriB"

Partie analyse de code (8 points)

Question 6 (1,5 points)

Ce code produit des affichages.
Donnez ces affichages dans le
tableau suivant :

```
1 l = [1, 2, 3]
2 t = (4, 5, 6)
3 for i in range(len(t)) :
4     l.insert(t[i] % 3, t[i])
5     print(f"i = {i} : l = {l}")
```

affichages

Question 7 (1,5 points)

a) Quel affichage produit ce code ?

b) Quel nombre faudrait-il à la place de **147**
pour afficher le résultat **3120** ?

```
1 def f(n) :
2     c = ""
3     x = n
4     while x >= 1 :
5         c = str(x % 4) + c
6         x = x // 4
7     return c
8
9 print(f(147))
```

Question 8 (1,5 points)

a) Quel affichage produit ce code ?

b) Quelle liste faudrait-il à la place de **[3, 1, 5, 2, 4, 0]**
pour afficher **[0, 1, 2, 3, 4, 5]** ?

```
1 a = []
2 b = [3, 1, 5, 2, 4, 0]
3
4 for i in range(6) :
5     j = min(i, len(b) - 1)
6     a.append(b.pop(j))
7
8 print(a)
```

Question 9 (sur 2 points)

Après exécution de ce programme, indiquez la valeur de a, b, c et d ainsi que leur type. La cohérence des réponses sera prise en compte, en cas d'erreur intermédiaire.

```
1 a = str(40 // 3 + 5 % 2 / 4)
2 b = float(a[1 : -1] + a[len(a) - 1] + a[0])
3 c = int((1000 * b) % 100)
4 d = c % 3 == 0
```

variable	type	valeur
a		
b		
c		
d		

Question 10 (sur 1,5 points)

Donnez les 4 valeurs affichées
par le programme suivant

```
1 def f(a, b) :
2     if a == 0 or b < 0 :
3         return 0
4     else :
5         return b + f(a//2, b-3)
6
7 print(f(9,14))
8 print(f(10,6))
9 print(f(19,11))
10 print(f(8,17))
```

Réponses : $f(9, 14) =$ $f(10, 6) =$ $f(19, 11) =$ $f(8, 17) =$

Partie écriture de code (7 points)

Voici le début d'un code. Il montre un premier dictionnaire nommé **lesPrix** qui contient les prix de vente des articles d'une boulangerie. Les trois autres dictionnaires correspondent aux achats de trois clients. Par exemple le client 1 a acheté 4 baguettes et 1 brioche.

Il y a aussi la liste des clients de la journée, nommée **lesClients**.

```
lesPrix = {"baguette" : 1.15, "croissant" : 1.25, "brioche" : 3.60}
client1 = {"baguette" : 4, "brioche" : 1}
client2 = {"brioche" : 2, "baguette" : 5, "croissant" : 3}
client3 = {"baguette" : 3}
lesClients = [client1, client2, client3]
```

Question 11 (sur 1,5 point)

Complétez le code de la fonction **montant**, qui prend en paramètres **deux dictionnaires** (un pour le client, un pour les prix) et qui **retourne le montant total** dépensé par le client dans la boulangerie. Par exemple on pourra exécuter l'instruction **print(montant(client1, lesPrix))**. Dans cette version, vous **ne donnerez pas** un code en une ligne.

```
def montant(client, prix) :
```

Question 12 (sur 1,5 point)

Même question que la précédente, mais cette fois votre code doit tenir en une seule ligne.

```
def montant(client, prix) :
```

Question 13 (sur 2 points)

Complétez le code de la fonction **chiffreJournée**, qui prend en paramètres une liste de clients et un dictionnaire de prix, et qui retourne le chiffre total réalisé par la boulangerie au cours de la journée où sont passés les clients de la liste. Vous pourrez donner une version en une ligne ou en plusieurs lignes, comme vous voulez.

```
def chiffreJournée(clients, prix) :
```

Question 14 (sur 2 points)

Complétez le code de la fonction **ajouterClient**, qui prend en paramètres une liste de clients et un dictionnaire de prix, et qui crée un client (dictionnaire) en demandant, pour chaque article de la boulangerie, la quantité achetée par le client. A la fin, ce client est ajouté à la liste des clients.

```
def ajouterClient(clients, prix) :
```