

Algorithmique et programmation 1

CC2 - Lundi 7/11/2022 de 9h30 à 10h30 en A4

- Aucun document n'est autorisé, ni ordinateurs, tablettes, calculatrices et/ou téléphones portables.
- L'examen est noté sur 20 points, le barème est donné à titre indicatif.
- Les programmes doivent être écrits en Python 3, en utilisant uniquement les notions vues en cours. Par conséquent, **l'utilisation des fonctions** (autres que celles vues en CM/TD), **des tableaux et des chaînes de caractères** (autre que pour l'affichage) ... ou toute autre spécificité du langage est interdite. Le non respect de cette consigne entraînera la note de 0 à la question concernée.
- Pour les **exercices 2 et 4 uniquement**, la réponse doit contenir une entête incluant le rôle et la liste des variables (nom et type).

Exercice 1. Questions de cours

1 + 1 + 1 + 2 = 5 pts

- 1. J'ai perdu mon support de cours, et je ne me souviens plus de la définition complète d'un algorithme.

Un algorithme est une ··· (1) ··· ··· (2) ···, ··· (3) ··· et ··· (4) ··· d'opérations ou d'instructions que l'on applique à un nombre fini de données, et qui permet de résoudre un problème.

Compléter cette définition.

- 2. Donner la définition d'une variable et d'une affectation. Illustrer sur un exemple.

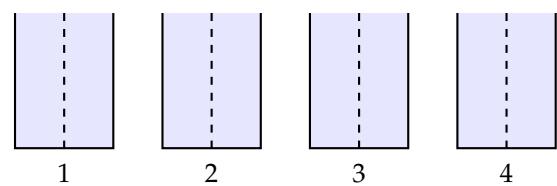
- 3. Parmi ces instructions, la(les)quelle(s) permet(tent) de lire une valeur au clavier et de la stocker dans la variable `a` ?

- (a) `a = input()`
- (b) `int(input("a ="))`
- (c) `print(a)`

Pour la ou les autre(s) instruction(s), expliquer ce qu'elle(s) permet(tent) de faire.

- 4. Recopier proprement les quatre schémas de droite. Sur chacun d'eux, donner la valeur de l'ensemble de toutes les variables créées après les instructions aux lignes 1 à 4. En particulier, sur le schéma *i*, noter les variables (nom et valeur) créées par les instructions aux lignes 1 à *i* (incluses), pour $i \in \{1, \dots, 4\}$.

```
1 v1 = 18 / 3
2 v2 = 4 - v1
3 v1 = 3
4 v3 = v1 ** 2
5 print("v1=", v1, "et v2=", 'v2', "et v3=", v3)
```



Quel est enfin le résultat de l'exécution de l'instruction à la ligne 5 ?

Exercice 2. Triplets pythagoriciens

1 + 1.5 = 2.5 pts

Un triplet pythagoricien est un triplet (x, y, z) d'entiers naturels non nuls vérifiant la relation de Pythagore ci-dessous :

$$x^2 + y^2 = z^2.$$

- 1. Le triplet $(x, y, z) = (3, 4, 5)$ est-il un triplet pythagoricien ?
- 2. Écrire un programme qui permet de déterminer si un triplet (x, y, z) de valeurs entières lues au clavier est un triplet pythagoricien.

Exercice 3. Structures conditionnelles imbriquées

1 + 1 + 1 = 3 pts

- 1. Quel est le résultat de l'exécution du programme Python suivant pour X = 65 ?

```
var = (X / 5) - ((X - (X % 5)) // 5)
if var == 0:
    print("A", end="")
    if (X / 5) == (X // 5):
        print("B")
    else:
        print("C")
else:
    print("D", end="")
    if (X / 5) != (X // 5):
        print("E")
    else:
        print("F")
```

- 2. Donner une valeur positive pour laquelle le résultat de l'exécution est DE.
► 3. Le programme contient-il une ou plusieurs lignes (instructions) qui ne pourront jamais être exécutée(s), quelque soit la valeur de la variable X ? Si oui, la(les)quelle(s) et pourquoi ?

Exercice 4. Triangle équilatéral, isocèle ou quelconque

1 + 3 = 4 pts

Soit un triangle de sommets A, B et C. Ce triangle est

$\begin{cases} \text{équilatéral} & \text{si les 3 côtés ont la même longueur, avec longueur } AB = \sqrt{(x_B - x_A)^2 + (y_B - y_A)^2}, \\ \text{isocèle} & \text{si deux côtés seulement ont la même longueur,} \\ \text{et quelconque} & \text{si les 3 côtés sont de longueur différente.} \end{cases}$

- 1. Quel est le type du triangle passant par les points A = (-1, 0), B = (1, 1) et C = (-1, -1) et pourquoi ?
► 2. Écrire un programme qui lit les coordonnées réelles (x_A, y_A) , (x_B, y_B) et (x_C, y_C) des points A, B, C, respectivement, puis qui détermine et affiche si le triangle est équilatéral, isocèle ou quelconque.

Exercice 5. Structures itératives

1 + 1 + 1 = 3 pts

- 1. Donner le résultat de l'exécution du programme ci-dessous pour $n = 9$ et $n = 10$.

```
i = 1
while i != n:
    print(i, ', ', end='')
    i = i + 2 * i
```

- 2. En s'inspirant de ce programme, proposer un nouveau programme qui lit une valeur entière n et qui affiche les valeurs entières divisibles par 2 entre 0 et n (inclus), en utilisant une boucle **while**.
► 3. Modifier ce nouveau programme pour utiliser une boucle **for**.

Exercice 6. Programme mystère

1.5 + 1 = 2.5 pts

Un étudiant vient de m'envoyer le programme suivant, sans aucune autre explication.

```
n = int(input("n= "))
for i in range(0, n+1, 1):
    for j in range(0, n-i, 1):
        print(" ", end=' ')
    print("*", end=' ')
    for j in range(0, i, 1):
        print(" ", end=' ')
    for j in range(0, i, 1):
        print(" ", end=' ')
    print("*")
```

La seule chose, après avoir passé des heures et des heures à essayer de le comprendre, il ne sait visiblement toujours pas ce qu'il fait, et il me demande de l'aide.

- 1. Que fait le programme pour $n = 4$? Et pour $n = 5$?
► 2. Réécrire le programme en utilisant que des boucles *while*.