

# Algorithmique 1

CT1 - Mardi 15/12/2020 de 8h30 à 10h30

**Numéro d'étudiant :** ..... **Note :** .....

- Indiquer le numéro d'étudiant sur chacune des feuilles du sujet.
  - Aucun document n'est autorisé, ni aucun matériel électronique (ordinateurs, calculatrices et/ou téléphones portables).
  - L'examen est noté sur 23 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, ... ou tout autre spécificité du langage est interdite.
  - Pour les **exercices 3, 5 et 7**, chaque réponse doit contenir une entête incluant le rôle et la liste des variables (nom et type).

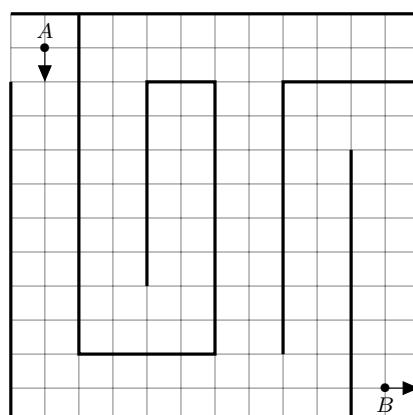
## Exercice 1. Une tortue dans un labyrinthe

1 pt

Votre tortue est perdue dans un labyrinthe. Mais heureusement, vous lui avez appris à communiquer avec vous, sur commande. Maintenant, elle sait :

- avancer d'un certain nombre de pas,
  - et tourner d'un certain angle dans le sens des aiguilles d'une montre (sens négatif).

- 1. Donner la suite d'instructions (où une case correspond à un pas) pour la faire sortir du labyrinthe, c'est-à-dire, la faire aller du point A (dans la direction indiquée) au point B (dans la direction indiquée), sans jamais la faire revenir sur ces pas.



.....  
.....  
.....  
.....  
.....

## Exercice 2. Questions de cours

$$1 + 1 + 1 = 3 \text{ pts}$$

- 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.

(1) .....

(2) .....

(3) .....

(4) .....

- 2. Donner une définition de la notion de *variable*, et une de celle d'*affectation*.

- 3. Quel est le résultat de l'exécution du programme suivant, pour  $a = 19$ ?

```
a = int(input("a= ")) # valeur lue : 19
b = a // 3
c = a % 3
d = b * 3 + c
b = (a == d)
print("a = ", a, " , b = ", b, " , c = ", c, " et d = ", d)
```

.....  
.....

### **Exercice 3. Vive le confinement**

$$1.5 + 1 = 2.5 \text{ pts}$$

Pendant le confinement, vous êtes devenu un vrai boulanger. Maintenant, vous savez que pour 3 baguettes, vous avez besoin de 500g de farine, 300g d'eau, 15g de levure de boulanger, et 2 cuillères à café de sel.

- 1. Écrire un programme qui lit le nombre de baguettes que vous voulez cuire, puis qui détermine et affiche les proportions pour chacun des ingrédients.

Argh, il ne vous reste plus que 1.2kg de farine dans votre placard.

- 2. Que faut-il ajouter à votre programme pour indiquer si vous aurez suffisamment d'ingrédients pour cuire le nombre de baguettes que vous souhaitez.

.....

.....

.....

.....

.....

.....

.....

#### Exercice 4. Structures conditionnelles

1 + 1 + 1 = 3 pts

Soit le programme Python suivant.

```
if var <= 17:  
    if var >= 0:  
        print("... var est dans l'intervalle [0,17]")
```

- 1. Réécrire ce programme pour ne plus utiliser de structures conditionnelles imbriquées, mais une seule structure conditionnelle.

```
.....  
.....  
.....  
.....
```

- 2. Quels sont les résultats des exécutions du programme suivant, en donnant, à la variable entière *var*, les valeurs successives suivantes : 37, 61, 120, et 2020.

```
var = int(input("Entrez une valeur: "))  
if var < 120:  
    print("A", end="")  
    if var >= 120:  
        print("B")  
    elif var >= 60:  
        print("C")  
    else:  
        print("D")  
else:  
    print("E", end="")  
    if var < 150:  
        print("F")  
    else:  
        print("G")
```

```
.....  
.....  
.....  
.....  
.....  
.....  
.....  
.....
```

- 3. Le message "AB" peut-il être affiché ? Si oui, donner une valeur de *var* permettant d'afficher ce message. Sinon proposer une explication.

```
.....  
.....  
.....  
.....  
.....
```

### **Exercice 5. Le prix à payer pour des photos**

1.5 pts

Un magasin de photos propose le développement au tarif de 0.16 euros l'unité. Le tarif est réduit à 0.12 euros l'unité pour une commande d'au moins 75 photos.

- 1. Écrire un programme qui lit le nombre de photos à développer, puis qui détermine et affiche la somme à payer en euros.

## Exercice 6. Structures itératives

$$1.5 + 1.5 = 3 \text{ pts}$$

- 1. Le programme suivant est-il erroné? Pour aider à répondre, donner le résultat de son exécution pour  $n = -1$ ,  $n = 0$ ,  $n = 1$ , et  $n = 2$ . Quelle est alors la conclusion?

```
n = int(input("Valeur de n: "))
i = 0
while i != n:
    print(i)
    i = i + 2
```

.....

.....

.....

.....

.....

- 2. En s'inspirant de ce programme, et de sa version éventuellement corrigée, proposer un programme qui lit une valeur  $n$  et qui affiche les valeurs entières paires entre 0 et  $n$  (inclus), par ordre décroissant et en utilisant une boucle **while**.

## Exercice 7. Horaires de bus

1.5 pts

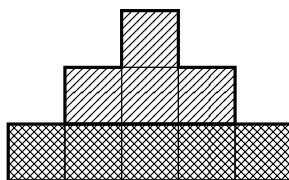
À Perpignan, le premier bus 4 passe à l'arrêt "Place de la Catalogne" à 6h du matin, le dernier à 23h, et un bus passe toutes les 20min.

- 1. Écrire un programme qui affiche tous les horaires de passage du bus 4 à cet arrêt sur une journée, en utilisant une boucle **while**.

### **Exercice 8. Jouons aux cubes**

$$2 + 1.5 = 3.5 \text{ pts}$$

On souhaite connaître le nombre de cubes nécessaires pour construire une pyramide de hauteur  $n$ , comme celle donnée ci-dessous pour  $n = 3$ .



Si on note  $S(n)$  le nombre de cubes pour une pyramide de hauteur  $n$ , on a :

$$S(n) = S(n - 1) + (2 \cdot n - 1), \quad \text{avec} \quad S(0) = 0.$$

Par exemple,  $S(3) = S(2) + 5 = 9$ .

- 1. Écrire un programme qui lit la hauteur  $n$  de la pyramide, puis qui calcule de manière itérative, en utilisant une boucle **while**, et affiche le nombre de cubes nécessaire.

- 2. Peut-on écrire ce programme avec une boucle **for**. Si oui, donner le programme. Sinon pourquoi?

.....

.....

.....

.....

.....

.....

.....

.....

.....

### Exercice 9. Tableau

$$2 + 2 = 4 \text{ pts}$$

Soit  $t$  un tableau d'entiers de taille  $n$ . Ce tableau contient  $k$  entiers positifs ou nuls, avec  $k \in \{0, \dots, n\}$ , puis  $n - k$  entiers égaux à  $-1$ , comme le tableau ci-dessous pour  $(n, k) = (8, 5)$  :

$$t = \{1, 3, 0, 2, 5, -1, -1, -1\}.$$

- 1. Écrire un programme qui lit le contenu du tableau, et qui vérifie si il respecte la spécification ci-dessus.

- 2. Que faut-il ajouter à ce programme pour lire un élément (entier) et l'insérer correctement dans le tableau ?

.....

.....

.....

.....

.....

.....

.....

.....

.....