



# Exercices Algorithmique



## Table des matières

<i>Section 1 Déroulement.....</i>	<i>3</i>
<i>Section 2 Numération binaire.....</i>	<i>4</i>
<i>Section 3 Installation de Python – Affectation de variables.....</i>	<i>5</i>
<i>Section 4 Installation de Wing IDE ou VS Code – Lecture clavier et affichage.....</i>	<i>7</i>
<i>Section 5 Codage de Tests.....</i>	<i>10</i>
<i>Section 6 Codage de Boucles TantQue.....</i>	<i>14</i>
<i>Section 7 Codage de Boucles Pour.....</i>	<i>16</i>
<i>Section 8 Codage de Tableau.....</i>	<i>17</i>
<i>Section 9 Codage de Fonctions.....</i>	<i>19</i>
<i>Section 10 Utilisation de fichiers.....</i>	<i>20</i>



## Section 1 Déroulement

---

Les exercices suivants vont être faits au cours de la semaine.

Aux moments où je le demanderai, les exercices seront à envoyer à l'adresse mail suivante :

[contact@pragma-tec.fr](mailto:contact@pragma-tec.fr)

Chaque fichier envoyé sera composé du no de l'exo suivi de votre nom

Exemple

exo4Dupond.py

L'ensemble des fichiers sera placé dans un .zip (pas d'autre format) et portera votre nom

ex : Dupond.zip

Ceci est un travail personnel : vous pouvez échanger des idées mais je ne dois pas trouver 2 fois le même code !



## Section 2 Numération binaire

---

### 2.1 But

- Conversion de nombres en binaire, décimal, hexadécimal

### 2.2 Enoncé

Convertir les nombres décimaux suivants en nombres binaires codés sur un octet (sans utiliser le Web SVP).

Vos résultats sont à placer dans le fichier Word ou txt de nom `exo2<votreNom>.doc` ou `.txt`

54

225

109

Convertir les nombres binaires suivants en décimal

101

01100011

11111110

Convertir les nombres hexa suivants en décimal **signé**

0x7E

0x81

0xFE



## Section 3 Installation de Python – Affectation de variables

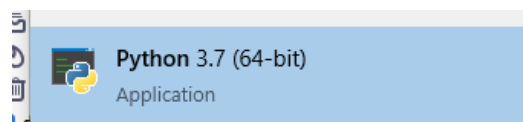
---

### 3.1 But

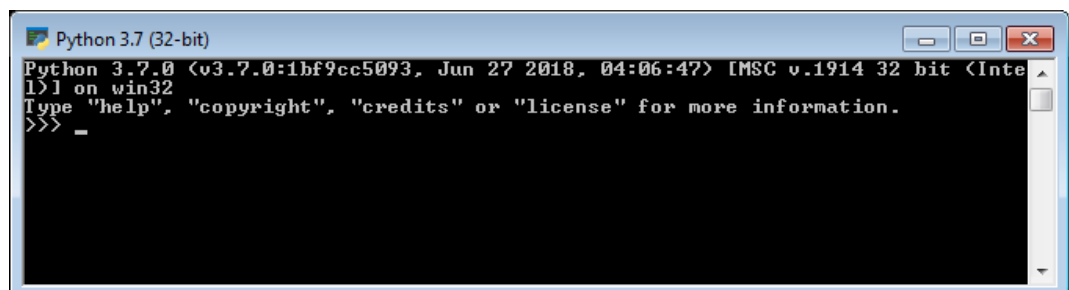
- Installer Python
- Quelques opérations

### 3.2 Enoncé

- Aller sur le site <https://www.python.org/>
- Sur Windows
  - Télécharger la dernière version 3.x.x
  - Installer
- Sur Mac
  - Télécharger et installer le fichier pkg
- Sur Linux
  - Python est pré installé. Vérifier la version avec `python -V`
  - Réaliser un download d'une version plus récente
  - <https://openclassrooms.com/en/courses/7168871-apprenez-les-bases-du-langage-python?archived-source=235344> pour la suite



- Lancer la console Python



- Réaliser les opérations suivantes :
  - $17 * 7$
  - $120 / 17$
  - $120 \% 17$



Le site suivant est très utile pour Python :

<https://www.w3schools.com/python/default.asp>

### 3.3 Utiliser des variables

En Python une variable :

- Ne doit pas commencer par un chiffre
- Composé de lettres minuscules ou majuscules, chiffres, '\_'
- Sensible à la casse : nom, Nom, NOM sont des variables différentes

Il n'y a pas de déclaration de type en Python : il le fait au mieux :

- `a=10` a contient un int ou un long en fonction de la taille
- `a=3.14` a contient un float
- `a="abcd"` a contient un str (string)
- `a=5+4j` a contient un complex
- `a=[1,5,6]` a contient une list
- .....
- La fonction `type(var)` fournit le type de la variable

L'affectation d'une valeur se fait par "="

Dans la fenêtre console taper :

- `a=5`
- `a`
- `type(a)`
- `b=3.14`
- `type(b)`
- `e = a+b`
- `e`
- `type(e)`
- `c="abcd"`
- `type(c)`
- `d=5+4j`
- `type(d)`



## Section 4 Installation de Wing IDE ou VS Code – Lecture clavier et affichage

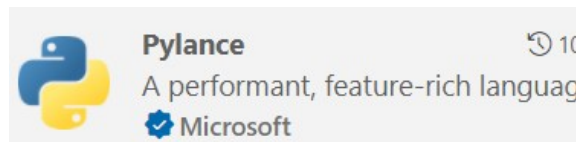
---

### 4.1 But

- Installer l'éditeur de texte Wing IDE ou Visual Studio Code
- Quelques opérations

### 4.2 Enoncé

- Nous pouvons utiliser l'éditeur de texte Wing IDE. Cet outil dédié à Python contient :
  - Debugger intégré
  - Aide en ligne
  - Aide à la saisie
- Installation de l'éditeur de texte Wing IDE :
  - Télécharger et installer l'outil, version Personal  
<https://wingware.com/downloads/wing-personal>
  - Le lancer
- Nous pouvons aussi utiliser Visual Studio Code. Y ajouter le plugin Pylance



-



En Python, la lecture d'une chaîne au clavier se fait par `input()`

Exemple :

```
nom = input("Saisir votre nom")
```

L'affichage d'une valeur se fait par `print()`

Exemple :

```
print(nom)
```

Créer sur votre machine le répertoire Documents/ExoPython (ou tout autre répertoire dédié aux exercices de la semaine)

Pour tous les exercices, si vous utilisez Wing IDE, les fichiers source commenceront par la ligne

```
# -*- coding: utf-8 -*-
```

Traduire les pseudo-code suivants en instructions Python et essayez les.

Nommer le fichier `exo4<votreNom>.py`

#### 4.2.1

Entier val, double

Début

Val  $\leftarrow$  231

Double  $\leftarrow$  Val \* 2

Ecrire Val

Ecrire Double

Fin

#### 4.2.2

Ecrire un programme qui demande un nombre à l'utilisateur, puis qui calcule et affiche le carré de ce nombre.

Indices :

Pour convertir la chaîne « 123 » en nombre 123 écrire :

```
Val = int("123")
```

Pour élever A à la puissance B : `A ** B`





#### 4.2.3

Ecrire un programme qui lit le prix HT d'un article, le nombre d'articles et le taux de TVA, et qui fournit le prix total TTC correspondant.

Faire en sorte que des libellés apparaissent clairement. Ex

Prix total 10.30 €



## Section 5 Codage de Tests

---

### 5.1 But

- Utilisation du SI

### 5.2 Enoncé

Créer le fichier `exo5<votreNom>.py`

Soit le pseudo code suivant :

```
SI age >= 18
ALORS
    Afficher "Vous êtes majeur"
SINON
    Afficher "Vous êtes mineur"
FINSI
```

Ce pseudo-code se traduit en Python de la façon suivante :

```
if age >= 18 :
    print("Vous êtes majeur")
else:
    print("Vous êtes mineur")
```

dans ce qui suit, écrire vos **pseudo-codes** avant le code Python dans `Exo5_<votreNom>.py` . Le pseudo code est à placer entre commentaires du langage python : soit `#` en début de chaque ligne ou un ensemble de lignes encadrés par 3 guillemets

Exemples

```
# ceci est un commentaire
```

```
#
```

```
""" commentaire sur plusieurs lignes
```

```
    suite du commentaires
```

```
"""
```



Ecrire un code Python qui demande un nombre à l'utilisateur, et l'informe ensuite si ce nombre est positif ou négatif

### 5.3

Ecrire un algorithme qui demande deux nombres à l'utilisateur et l'informe ensuite si leur produit est négatif ou positif (on laisse de côté le cas où le produit est nul). Attention toutefois : on ne doit **pas** calculer (et du coup tester) le produit des deux nombres.

### 5.4

Ecrire un algorithme qui demande trois noms à l'utilisateur et l'informe ensuite s'ils sont rangés ou non dans l'ordre alphabétique.

### 5.5

Le prix du ticket de cinéma dépend de l'âge, et de la détention d'une carte d'abonnement :

Sans carte : 10€ pour une personne de plus de 25 ans, 7.5€ sinon

Sur présentation de la carte une réduction de 20 % est faite

Calculer le prix du billet en fonction de ces paramètres.

### 5.6

Définition d'une année bissextile sur wikipédia :

Depuis l'[ajustement du calendrier grégorien](#), l'année sera bissextile (elle aura 366 jours)<sup>1</sup> :

- si l'année est divisible par 4 et non divisible par 100, ou
- si l'année est divisible par 400.
- Sinon, l'année n'est pas bissextile (elle a 365 jours).

(« divisible » signifie que la division donne un nombre entier, sans reste).

Ainsi, 2019 n'est pas bissextile. L'an [2008](#) était bissextile suivant la première règle (divisible par 4 et non divisible par 100). L'an [1900](#) n'était pas bissextile car divisible par 4, mais aussi par 100 (première règle non respectée) et non divisible par 400 (seconde règle non respectée). L'an [2000](#) était bissextile car divisible par 400.

Ecrire un code Python qui demande la valeur d'une année et indique si elle est bissextile. NE PAS RECOPIER LA SOLUTION disponible sur le web. On attend votre propre code.



## 5.7

Cet algorithme est destiné à prédire l'avenir, et il doit être infaillible !

Il lira au clavier l'heure et les minutes, et il affichera l'heure qu'il sera une minute plus tard. Par exemple, si l'utilisateur tape 21 puis 32, l'algorithme doit répondre :

"Dans une minute, il sera 21 heures 33".

NB : on suppose que l'utilisateur entre une heure valide. Pas besoin donc de la vérifier.

NB : il n'y aura pas de 's' à heure dans la réponse "il sera 1 heure 33"

## 5.8

Un magasin de reprographie facture 0,10 € les dix premières photocopies, 0,09 € les vingt suivantes et 0,08 € au-delà. Ecrivez un algorithme qui demande à l'utilisateur le nombre de photocopies effectuées et qui affiche la facture correspondante.

## 5.9

Les habitants de la principauté de Conamo paient l'impôt selon les règles suivantes :

- les hommes de plus de 20 ans paient l'impôt . Dans ce cas la valeur de l'impôt est de 10 % du revenu imposable entre 20 et 40 ans compris, 15 % au-delà.
- les femmes paient l'impôt si elles ont entre 18 et 35 ans . Dans ce cas la valeur de l'impôt est de 10 % du revenu imposable.
- les autres ne paient pas d'impôt

Le programme demandera donc l'âge et le sexe du Conamolien, et se prononcera donc ensuite sur le fait que l'habitant est imposable.

Il demandera le revenu imposable et affichera la valeur de l'impôt.



## 5.10

Une compagnie d'assurance automobile propose à ses clients quatre familles de tarifs identifiables par une couleur, du moins au plus onéreux : tarifs bleu, vert, orange et rouge. Le tarif dépend de la situation du conducteur :

- un conducteur de moins de 25 ans et titulaire du permis depuis moins de deux ans, se voit attribuer le tarif rouge, si toutefois il n'a jamais été responsable d'accident. Sinon, la compagnie refuse de l'assurer.
- un conducteur de moins de 25 ans et titulaire du permis depuis plus de deux ans, ou de plus de 25 ans mais titulaire du permis depuis moins de deux ans a le droit au tarif orange s'il n'a jamais provoqué d'accident, au tarif rouge pour un accident, sinon il est refusé.
- un conducteur de plus de 25 ans titulaire du permis depuis plus de deux ans bénéficie du tarif vert s'il n'est à l'origine d'aucun accident et du tarif orange pour un accident, du tarif rouge pour deux accidents, et refusé au-delà
- De plus, pour encourager la fidélité des clients acceptés, la compagnie propose un contrat de la couleur immédiatement la plus avantageuse s'il est entré dans la maison depuis plus d'un an.

Ecrire l'algorithme permettant de saisir les données nécessaires (sans contrôle de saisie) et de traiter ce problème. Avant de se lancer à corps perdu dans cet exercice, on pourra réfléchir un peu et s'apercevoir qu'il est plus simple qu'il n'en a l'air (cela s'appelle faire une analyse !)



## Section 6 Codage de Boucles TantQue

---

### 6.1 But

- Utilisation du TantQue

### 6.2 Enoncé

Créer le fichier `exo6_<votreNom>.py` . Mettre les pseudo-codes en tant que commentaires dans le même fichier

Soit le pseudo code suivant :

valeur  $\leftarrow$  0

TantQue valeur < 10

    Afficher valeur \* 4

    valeur  $\leftarrow$  valeur + 1

FinTantQue

Ce pseudo-code se traduit en Python de la façon suivante :

```
valeur=0
while valeur <= 10:
    print (valeur * 4)
    valeur +=1
```

Ecrire un algorithme qui demande à l'utilisateur un nombre compris entre 1 et 3 jusqu'à ce que la réponse convienne.

Ecrire le code Python.

### 6.3

Ecrire un algorithme qui demande un nombre compris entre 10 et 20, jusqu'à ce que la réponse convienne. En cas de réponse supérieure à 20, on fera apparaître un message : « Plus petit ! », et inversement, « Plus grand ! » si le nombre est inférieur à 10.

Ecrire le code Python.



## 6.4

Reprendre l'exercice de calcul d'année bissextile. Faire en sorte que l'on demande des dates jusqu'à la saisie de la lettre 'q', pour sortir. Pour chaque date saisie, afficher si la date est bissextile ou non.

## 6.5

Ecrire un algorithme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu'à ce nombre. Par exemple, si l'on entre 5, le programme doit calculer :

$$1 + 2 + 3 + 4 + 5 = 15$$

## 6.6

Lire la suite des prix (en euros entiers et terminée par zéro) des achats d'un client. Calculer la somme qu'il doit, lire la somme qu'il paye, et simuler la remise de la monnaie en affichant les textes "10 Euros", "5 Euros" et "1 Euro" autant de fois qu'il y a de coupures de chaque sorte à rendre.

## 6.7

Ecrire un algorithme qui demande successivement des nombres à l'utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces nombres. La saisie d'un 0 arrête la demande .

L'algo affiche le plus grand nombre et sa position. Exemple :

Entrez le nombre numéro 1 : 12

Entrez le nombre numéro 2 : 14

etc.

Entrez le nombre numéro 20 : 0

Le plus grand de ces nombres est : 14 en position 2

## 6.8

Lire la suite des prix (en euros entiers et terminée par zéro) des achats d'un client. Calculer la somme qu'il doit, lire la somme qu'il paye, et simuler la remise de la monnaie en affichant les textes "10 Euros", "5 Euros" et "1 Euro" autant de fois qu'il y a de coupures de chaque sorte à rendre.



## Section 7 Codage de Boucles Pour

---

### 7.1 But

- Utilisation du Pour

### 7.2 Enoncé

Soit le pseudo code suivant :

valeur  $\leftarrow$  0

Pour valeur = 1 à 10 Pas de 1

Afficher valeur \* 4

FinPour

Ce pseudo-code se traduit en Python de la façon suivante :

```
for valeur in range (1,11,1):  
    print (valeur * 4)
```

Créer le fichier `exo7_<votreNom>.py`. Compléter le `.word` avec les pseudo-codes.

Ecrire un algorithme qui demande un nombre de départ, et qui ensuite écrit la table de multiplication de ce nombre, présentée comme suit (cas où l'utilisateur entre le nombre 7) :

Table de 7 :

7 x 1 = 7

7 x 2 = 14

7 x 3 = 21

...

7 x 10 = 70

### 7.3

Ecrire un algorithme qui demande un nombre de départ, et qui calcule la somme des entiers jusqu'à ce nombre. Par exemple, si l'on entre 5, le programme doit calculer :

1 + 2 + 3 + 4 + 5 = 15





## Section 8 Codage de Tableau

---

### 8.1 But

- Utilisation de tableau

### 8.2 Enoncé

Créer le fichier `exo8_<votreNom>.py` Modifier le `.word` avec vos pseudo-codes

En Python les tableaux simples n'existent pas. Python utilise directement les listes, qui est plus général qu'un tableau.

Pour déclarer une liste vide:

```
liste=[ ]
```

Pour déclarer une liste et y mettre des valeurs :

```
liste = ["a", "d", "m"]
```

Pour ajouter un élément dans une liste :

```
liste.append("hello")
```

Pour atteindre un élément

```
liste[2]
```

Pour parcourir une liste uniquement pour son contenu

```
for x in liste:  
    print (x)
```

Pour parcourir une liste en utilisant l'index et contenu

```
for i,x in enumerate(liste):  
    print (x, " a l'index " , i)
```

Ecrire un algorithme qui déclare un tableau de 9 notes, dont on fait ensuite saisir les valeurs par l'utilisateur.

Calculer et afficher la somme des valeurs.



### 8.3

Ecrivez un algorithme constituant un tableau, à partir de deux tableaux de même longueur préalablement saisis. Le nouveau tableau sera la somme des éléments des deux tableaux de départ.

Tableau 1 :

4	8	7	9	1	5	4	6
---	---	---	---	---	---	---	---

Tableau 2 :

7	6	5	2	1	3	7	4
---	---	---	---	---	---	---	---

Tableau à constituer :

11	14	12	11	2	8	11	10
----	----	----	----	---	---	----	----

### 8.4

Ecrivez un algorithme permettant, à l'utilisateur de saisir les noms d'élèves et notes d'une classe. Le programme, une fois la saisie terminée (touche 'q'), affiche la moyenne puis les noms et notes des élèves dont la note est supérieure à la moyenne de la classe.



## Section 9 Codage de Fonctions

---

### 9.1 But

- Créer des fonctions et les utiliser

### 9.2 Enoncé

Créer le fichier `exo9_<votreNom>.py`

En Python l'écriture d'une fonction est comme suit :

```
function maFonction(param1, param2):  
    // utilisation des paramètres  
    return resultat // si nécessaire
```

L'utilisation s'écrit :

```
a = maFonction (12, 10)
```

Écrire un algorithme qui permette de connaître ses chances de gagner au tiercé, quarté, quinté et autres impôts volontaires.

On demande à l'utilisateur le nombre de chevaux partants, et le nombre de chevaux joués. Les deux messages affichés devront être :

Dans l'ordre : une chance sur X de gagner

Dans le désordre : une chance sur Y de gagner

X et Y nous sont donnés par la formule suivante, si n est le nombre de chevaux partants et p le nombre de chevaux joués (on rappelle que le signe ! signifie "factorielle") :

$$X = n! / (n - p) !$$
$$Y = n! / (p! * (n - p) !)$$

### 9.3

Écrivez une fonction qui renvoie le nombre de voyelles contenues dans une chaîne de caractères passée en argument. Au passage, notez qu'une fonction a tout à fait le droit d'appeler une autre fonction.

Essayer la fonction avec la chaîne "je m'amuse beaucoup en algorithmique"



## Section 10 Utilisation de fichiers

---

### 10.1 But

- Gérer des fichiers

### 10.2 Enoncé

Regarder le site [https://www.w3schools.com/python/python\\_file\\_write.asp](https://www.w3schools.com/python/python_file_write.asp) pour la gestion des fichiers.

Ecrire un algorithme puis code python pour la saisie d'un annuaire.

Le programme demande une suite de noms et no de téléphone jusqu'à la saisie d'un nom vide.

Les informations sont sauvegardées dans un fichier texte annuaire.txt.

Une nouvelle utilisation de `exo10Write_<votreNom>.py` provoque l'ajout en fin de fichier.

Dans `exo10Read_<votreNom>.py`

Ecrire un algorithme puis code python pour la recherche d'un no de téléphone à partir d'un nom. Le fichier annuaire.txt est utilisé en lecture.

Le programme boucle sur la saisie de noms jusqu'à chaîne vide.