

Les algorithmes sur lesquels on va travailler en classe doivent être traduits dans un langage de programmation pour être implantés dans un ordinateur.

Le langage choisi pour le lycée est Python.



A - Variables

Définition 1 : Un algorithme est une suite d'instructions élémentaires. Ces instructions s'appliquent dans un ordre déterminé à des données et fournissent en un nombre fini d'étapes des résultats.

Exemple 1 : Une recette de cuisine pour faire un gâteau
(données : ,
étapes : ,
résultat :)

Exemple 2 : Au collège, on a déjà utilisé les algorithmes avec les programmes réalisés avec le logiciel Scratch.

En voici un exemple ci-contre :

- Tester le programme ci-contre pour la valeur 3, puis pour la valeur -1.
- Reprendre la question précédente avec x .



Définition 2 : Dans le programme précédent, plusieurs lettres sont utilisées pour ranger les différents calculs réalisés. Chacune de ces lettres s'appelle une variable. C'est une sorte de boîte où l'on va ranger un nombre.

Ici, notre programme contient variables.

Dans l'exemple précédent, nos variables portent pour nom une lettre majuscule : A, B, ...

Il est possible d'appeler nos variables comme bon nous semble avec une ou plusieurs lettres majuscules et/ou minuscules.

Définition 3 : Il existe plusieurs types de variables :

- **entier** (nombre entier relatif) ;
- **flottant** (nombre décimal) ;
- **chaîne de caractères** (suite ordonnée de caractères, c'est-à-dire de lettres, chiffres ou symboles) ;
- **booléen** (variable qui ne prend que deux valeurs : vrai ou faux. Cette valeur est le plus souvent donnée par un test).

Exemple 3 :

- "seconde" est une chaîne de 7 caractères. On dit que c'est une chaîne de longueur 7.
- La variable a qui contient le test ($7 > 1$) est de type booléen. Sa valeur est : Vrai

Remarques :

- Les chaînes de caractères peuvent s'ajouter, se mettre bout-à-bout (on parle de concaténer les chaînes). Par exemple, "seconde"+"3" donne "seconde3"
- Dans Python,
 - le type entier se nomme **int** (pour *integer*)
 - le type flottant se nomme **float** (pour *floating-point*)
 - le type chaîne de caractères se nomme **str** (pour *string*)

Exercice A1 :

On souhaite créer un programme dans lequel apparaissent les variables ville, pluviometrie et ensoleillement donnant respectivement le nom de la ville considérée, sa pluviométrie (en cm par m² arrondi à 0,1 près) et le nombre de jours d'ensoleillement de l'année en cours.

Donner le type de chacune des variables.

Exercice A2 :

Dans un programme permettant de tester si un triangle est isocèle, équilatéral, rectangle ou quelconque, apparaissent 4 variables : nom, longueur1, longueur2, longueur3 prenant respectivement pour valeur le nom du triangle (par exemple ABC) et les longueurs de ses trois côtés.

Donner le type de chacune des variables.

B - Affectation - Entrées et sorties

1) Affectation d'une valeur à une variable

Définition 4 : Affecter 3 à une variable A, c'est donner à la variable A la valeur 3. (Si on représente une variable comme une boîte, affecter 3 à une variable A, c'est mettre 3 dans la boîte A). On écrit alors :

- $A \leftarrow 3$ en langage naturel
- $A = 3$ en langage python

Remarque : L'instruction : $A \leftarrow B$ signifie qu'on affecte à A la valeur contenue dans la variable B.

2) Entrer une valeur dans une variable

Définition 5 : Entrer une valeur dans la variable A, c'est demander à l'utilisateur quelle valeur il veut affecter à la variable A. On écrit alors :

- $A \leftarrow ?$ ou Saisir A en langage naturel
- $A = \text{float}(\text{input}(\text{"choisir une valeur"}))$ ou $A = \text{float}(\text{input}())$ en langage python

Remarque : La commande input() attend ici que l'on rentre un nombre et que l'on appuie sur la touche Entrée. Mais ce nombre est d'abord saisi en tant que chaîne de caractères, puis converti en nombre « flottant » grâce à la fonction float ou en nombre entier grâce à la fonction int. Ce nombre sera alors affecté à la variable A.

3) Afficher une variable

Définition 6 : Afficher une variable A, c'est afficher (sur l'écran de la calculatrice, de l'ordinateur, ...) la valeur contenue dans A. On écrit alors :

- Afficher A en langage naturel
- `print(A)` en langage Python

Remarque :

La commande `print` permet d'afficher ce que l'on veut : que ce soit un nombre, une chaîne de caractères ou une affectation de variables.

`print("la réponse est")` affiche : la réponse est

`print("la réponse est", a)` affiche : la réponse est *valeur de a*

4) Calculs et opération dans Python

Dans le tableau ci-après, sont présentés les symboles utilisés pour les opérations de base.

Opérations	Symboles	Exemples
addition	+	2 + 5 donne 7
soustraction	-	8 - 2 donne 6
multiplication	*	6 * 7 donne 42
exponentiation (puissance)	**	5 ** 3 donne 125
division	/	7 / 2 donne 3.5
reste de division entière	%	7 % 3 donne 1
quotient de division entière	//	7 // 3 donne 2

Attention : Pour tester si une égalité est vraie ou fausse on utilise `==`.

Exemples 4 :

```
>>> a = 13
>>> print(a)
13
>>> a == 13
True
```

```
>>> var = 12.5
>>> var
12.5
>>> var == 49.3
False
```

```
>>> C = 3.8
>>> C = C + 1
>>> print(C)
4.8
```

- La ligne 1 ne donne aucun affichage. Si la variable **a** n'existait pas, alors elle est créée et prend la valeur 13. Et si elle existait déjà, cette variable prend aussi la valeur 13 (l'ancienne valeur est alors perdue).
- On remarque au passage que, dans l'interpréteur interactif, taper juste **var** a le même effet que **print(var)**. Ce n'est pas le cas dans l'éditeur où l'on a besoin de l'instruction **print** pour afficher un résultat.
- Il faut bien faire attention à ne pas confondre l'égalité mathématique **C = C + 1** qui est toujours fausse (ou l'équation d'inconnue C qui n'a pas de solution) avec la nouvelle affectation de la variable C à l'aide de l'ancienne valeur. Ce que l'on peut schématiser ainsi :

$$C_{\text{nouveau}} = C_{\text{ancien}} + 1$$

Exercice B1 :

1)

Algorithme	Python
$x \leftarrow ?$ $a \leftarrow 6 * x$ $b \leftarrow a + 10$ $c \leftarrow b/2$ Afficher c	<pre>x=float(input("x=")) a=6*x b=a+10 c=b/2 print(c)</pre>

Quelle est la valeur obtenue en sortie lorsqu'on entre :

- la valeur 2 ?
- la valeur - 4 ?

2)

Algorithme	Python
$n \leftarrow ?$ $p \leftarrow n + 1$ $q \leftarrow p + 1$ $r \leftarrow q + 1$ $s \leftarrow n^2 - p^2 - q^2 + r^2$ Afficher s	<pre>n=float(input("n=")) p=n+1 q=p+1 r=q+1 s=n**2-p**2-q**2+r**2 print(s)</pre>

Quelle est la valeur obtenue en sortie lorsqu'on entre :

- la valeur 0 ?
- la valeur 2 ?

Exercice B2 :

Voici un programme de calcul :

- Prendre un nombre
- Lui ajouter 8
- Multiplier le résultat par 3
- Enlever 24
- Enlever le nombre de départ

Mettre ce programme de calcul sous forme d'algorithme en langage naturel puis le traduire en Python

Algorithme	Python

Exercice B3 :

Ecrire un algorithme en langage naturel et en Python qui permet de calculer l'aire d'un rectangle quand on entre la longueur L et la largeur l.

Rappel :

Pour un rectangle de longueur L et de largeur l, l'aire est donnée par

Algorithme	Python

Exercice B4 :

On rappelle que le volume d'un cylindre est donné par la formule $V = \pi R^2 h$ où R est le rayon de la base et h la hauteur du cylindre.

Ecrire un algorithme, en langage naturel et en Python, qui permet d'obtenir le volume à partir de la hauteur et de rayon.

Algorithme	Python

C - Fonctions

Pour écrire un algorithme, on privilégiera l'utilisation d'une ou plusieurs fonctions.

On peut ensuite utiliser la fonction dans un programme ou directement dans la console de Python.

L'utilisation des fonctions évitent l'utilisation des « ←? » (Saisir) en début de programme et « Afficher » en fin.

Cela permettra de programmer plus rapidement.

Définition 7 : Une fonction est un bloc d'instructions qui a reçu un nom, dont le fonctionnement dépend d'un certain nombre de paramètres (les arguments de la fonction) et qui renvoie un (ou plusieurs) résultats via le mot clé Retourner. Elle se présente ainsi :

Langage naturel	Python
Fonction Nom(arguments) instructions Retourner	def Nom(arguments) : instructions return (résultat)

Remarques : Si l'on décortique la ligne de définition de la fonction, on trouve dans l'ordre :

- **def**, mot-clé qui est l'abréviation de « define » (définir, en anglais) et qui constitue le prélude à toute construction de fonction.
- Le nom de la fonction, qui se nomme exactement comme une variable. Il ne faut pas utiliser un nom de variable déjà instanciée pour nommer une fonction.
- La liste des paramètres qui seront fournis lors d'un appel à la fonction. Les paramètres sont séparés par des virgules et la liste est encadrée par des parenthèses ouvrante et fermante (là encore, les espaces sont optionnels mais améliorent la lisibilité).
- Les deux points qui clôturent la ligne.

Attention : Les parenthèses sont obligatoires, quand bien même votre fonction n'attendrait aucun paramètre.

Exemple 5 :

On peut définir une fonction qui calcule l'aire d'un triangle à partir de sa base et sa hauteur.

```
def aire_triangle (base, hauteur):  
    a=base*hauteur/2  
    return(a)
```

Dans la console, on peut calculer l'aire d'un triangle de base 8 et hauteur 7

```
Console Python  
  
*** Console de processus distant Réinitialisée ***  
>>>  
>>> aire_triangle(8,7)  
28.0
```

Exemple 2 : Voici deux fonctions appelées b et message

```
def b():  
    return("Bonjour")  
  
def message(prenom):  
    a=b()+prenom  
    return(a)
```

En saisissant b()+b() dans la console, on obtient
BonjourBonjour

On peut améliorer en ajoutant un espace

Si on saisit message("Bob")

Et avec un espace ...

```
Console Python  
  
>>> b()+b()  
'BonjourBonjour'  
>>> b()+" " +b()  
'Bonjour Bonjour'  
  
>>> message("Bob")  
'BonjourBob'  
  
>>> message(" Bob")  
'Bonjour Bob'
```

Remarques :

- Quand une fonction renvoie une valeur, elle le fait de façon exclusive : son déroulé est stoppé et plus aucune instruction de la fonction n'est évaluée.
- On peut utiliser une fonction durant l'exécution d'une autre fonction.

Exercice C1 :

En prévision des soldes, un commerçant s'apprête à modifier ses étiquettes.

1) Quel est le prix d'un article qui coûtait 40€ et qui est soldé à 30%?

2) Voici la fonction qu'il a écrit en Python :

```
def solde(prix,t):  
    p= prix*(1-t/100)  
    return(p)
```

a) Que représente la variable t ?

b) Que doit saisir le commerçant dans la console de Python pour obtenir le prix d'un article qui coûte 55€ et qui est soldé à 60% ? Quel est le prix de l'article soldé ?

Exercice C2:

Le prix de l'entrée dans un parc aquatique est 37€ pour un adulte et 28€ pour un enfant.

1) Programmer une fonction prix_parc dont les arguments sont le nombre d'adultes et le nombre d'enfants d'un groupe et qui retourne le prix payé par le groupe.

2) Tester votre fonction avec les valeurs suivantes :

Adultes	2	3	4	5
Enfants	3	2	5	4
Total				

A retenir : Il faut absolument respecter l'ordre des arguments.

Exercice C3:

1) On rappelle que le volume d'une pyramide est donné par la formule :

$$V = \frac{1}{3} \times \text{base} \times \text{hauteur}$$

Ecrire une fonction `vol_pyr` qui retourne le volume d'une pyramide dont on connaît l'aire de la base et la hauteur.

2) a) Compléter, en faisant appel à la fonction `vol_pyr`, le programme de la fonction `vol_pyr_rect` qui calcule le volume d'une pyramide à base rectangulaire à partir de la largeur `l` et de la longueur `L` de sa base ainsi que sa hauteur.

```
def vol_pyr_rect ( L, l, hauteur):
```

2) b) Tester votre fonction avec les valeurs suivantes:

Longueur	largeur	hauteur	Volume
6	2	4	
10	3	5	
2	5	4	

Exercice C4:

Rappel : Le quadrilatère ABCD est un parallélogramme si et seulement si ses diagonales

1) Compléter la fonction `milieu` ci dessous qui retourne les coordonnées du milieu M du segment [AB]

```
def milieu(xA,yA,xB,yB):
```

```
    xM =
```

```
    yM =
```

```
    return(xM,yM)
```

2) a) Compléter l'écriture de la fonction suivante pour qu'elle indique si ABCD est un parallélogramme.

```
def parall(xA,yA,xB,yB,xC,yC,xD,yD):
```

```
    return
```

2)b) Tester la fonction `parall` dans la console avec les points suivants :

A	B	C	D	Parallélogramme ?
(0; 0)	(2; 0)	(2; 3)	(0; 3)	
(1; 0)	(4; -3)	(-7; 5)	(6; 2)	
(1; 1)	(4; 3)	(9; 1)	(6; -1)	

Exercice C5 :

Rappel : un point M est sur la médiatrice de [AB] si et seulement si

1) Compléter la fonction `dist` ci-dessous qui retourne la distance AB.

```
from math import*
```

```
def dist(xA,yA,xB,yB):
```

```
    d =
```

```
    return d
```

2) a) Compléter l'écriture de la fonction suivante pour qu'elle indique si M est sur la médiatrice de [AB].

(*Rappel*: pour tester l'égalité de résultat, on utilise `=`)

```
def med(xM,yM,xA,yA,xB,yB):
```

```
    return
```

2)b) Tester la fonction `med` dans la console avec les points suivants :

M	A	B	M sur la médiatrice ?
(-5; -5)	(2; 0)	(0; 2)	
(1; 0)	(4; -3)	(-7; 5)	
(2; -1)	(1; 3)	(6; 0)	

Exercice C6 :

1) Créer une fonction , avec Python, qui calcule le coefficient directeur de la droite (AB) connaissant les coordonnées des points A et B, après avoir vérifié que les points n'aient pas la même abscisse.

2) Créer une fonction , avec Python, qui détermine si deux droites non verticales sont parallèles ou non, en connaissant les équations réduites des deux droites.

3) Utiliser les deux fonctions précédentes pour créer une fonction qui teste si des droites (AB) et (CD) sont parallèles ou non, à partir des coordonnées des quatre points.

D - Instruction conditionnelle : Le « Si »

Définition 8 : En algorithmique, on est très souvent amené à effectuer des instructions sous certaines conditions. On peut utiliser les relations de comparaison suivantes : $<$, $>$, \leq , \geq , $=$, \neq en langage natutel.

Langage naturel	Python
Si ... Alors instructions	if condition : instructions
Sinon instructions	else : instructions
FinSi	

Remarque :

- L'instruction "sinon" n'est pas obligatoire. Dans ce cas, l'algorithme ne fait rien si la condition n'est pas vérifiée.
- Il ne faut pas oublier les **deux-points** après la condition et après "else" en langage python.
- L'indentation (décalag) est **obligatoire** et se fait automatiquement sous Python (si on n'a pas oublié les deux-points)
- On supprime l'indentation pour sortir de l'instruction conditionnelle et poursuivre l'algorithme.

Exemple 6 :

1.	Entrée $A \leftarrow ?$
2.	Traitement $B \leftarrow A + 3$
3.	$C \leftarrow B \times 4 + 5$ Si $C > 0$ Alors
4.	$B \leftarrow \sqrt{C}$
5.	Sinon $B \leftarrow C^2$
6.	FinSi $D \leftarrow B \times 3$
	Sortie Afficher D

- Que contiennent les variables à la fin de cet algorithme si :
 - on saisit la valeur 52 dans A.
 - on saisit la valeur -10 dans A.

Pour répondre, on utilisera les deux tableaux ci-dessous :

Ligne	A	B	C	D
1				
2				
3				
4				
5				
6				

Ligne	A	B	C	D
1				
2				
3				
4				
5				
6				

- Quel est l'affichage obtenu dans ces deux cas ?

Voici une liste des opérateurs que l'on peut utiliser :

- $x == y$ est vrai quand x est égal à y,
- $x != y$ est vrai quand x est différent de y,
- $x > y$ est vrai quand x est strictement supérieur à y,
- $x < y$ est vrai quand x est strictement inférieur à y,
- $x \geq y$ est vrai quand x est supérieur ou égal à y,
- $x \leq y$ est vrai quand x est inférieur ou égal à y.

Remarque : Grâce à Python on peut vérifier une « double égalité » comme en mathématique : $a \leq b \leq c$ s'écrirait : $a \leq b \leq c$ ou on peut utiliser la commande **and** : $a \leq b$ and $b \leq c$ (plus long)

Exercice D1 :

Un site internet de développement de photos propose le tirage sur papier des photos à 0,11€ l'unité; le tarif passe à 0,08 € l'unité pour une commande d'au moins 100 photos. On veut créer un algorithme donnant le montant dépensé pour un nombre n de tirages.

- Si le nombre n de photos est strictement inférieur à 100,
le prix d'un tirage est :

le montant M dépensé est donc :

.....

- Si le nombre n de photos est supérieur ou égal à 100,
le prix d'un tirage est :

le montant M dépensé est donc :

.....

Compléter l'algorithme ci-dessous et observer sa traduction en Python

Algorithme	Python
n ← ?	n=int(input("Entrer n"))
Si	if n <100 :
alors M ←	M= 0.11*n
Sinon M ←	else:
Fin Si	M=0.08*n
Afficher "le montant est : "	print("le montant est", M)
Afficher M	

Exercice D2 :

On donne le programme suivant écrit en langage Python:

```
x= float(input("X = "))
if X<=250:
    C=75
else:
    C = 75+(X-250)*0.28
print(C)
```

- Qu'affiche le programme en sortie si on saisit en entrée :

a) X = 162 ?

b) X = 625 ?

- Une agence de location de voitures utilise le programme ci-dessus afin de calculer le coût C de la location d'une voiture pour X kilomètres parcourus.

Compléter les informations manquantes :

★ Tarif de la location :€

Ce tarif comprend la location de la voiture et permet de parcourir km)

★ Chaque kilomètre supplémentaire coûte €

Exercice D3 :

Un club sportif fait fabriquer des t-shirts au nom du club. Chaque t-shirt est facturé 4€, mais ils sont facturés 3,50 € si la commande est d'au moins 50 unités du produit. Soit N la variable égale au nombre de t-shirts commandés et P le prix à payer.

1a) Quel est le prix à payer pour 45 t-shirts ?

.....

.....

.....

.

1b) Quelle est la valeur de P si N = 120 ?

.....

.....

.....

.

2) Si le nombre N est strictement inférieur à 50 :

- le prix d'un t-shirt est
- le montant dépensé est

Si le nombre N est supérieur ou égal à 50 :

- le prix d'un t-shirt est
- le montant dépensé est

3) Compléter l'algorithme suivant et le traduire en Python :

Algorithme	Python.
$N \leftarrow ?$ si N Alors $P \leftarrow \dots$ Sinon $P \leftarrow \dots$ FinSi Afficher P	

4) Tester votre algorithme avec les valeurs suivantes.

Nombre de t-shirts	30	45	50	120	254
Montant					

Exercice D4 :

Un groupe de personnes souhaite réserver un chalet à la montagne pour skier. La location du chalet est de 800€ . Pour skier, le forfait à la semaine est de 220€ par personne. Mais, si le groupe est composé de 5 personnes ou plus, le forfait est à 180€ par personne.

1) Quelle est la dépense du groupe s'il est composé de 4 personnes ?

2) Quelle est la dépense du groupe s'il est composé de 6 personnes ?

3) a) Créer un algorithme qui affiche le prix payé par un groupe composé de n personnes puis le traduire en Python

Algorithme	Python

3)b) Vérifier les valeurs trouvées aux questions 1 et 2.

4) Modifier l'algorithme pour qu'il affiche la part que chaque membre du groupe doit payer.

Algorithme	Python

5) Tester votre algorithme pour les valeurs suivantes :

Nombre de personnes	3	5	7	12
Montant total				
Montant par personne				

Exercice D5 :

Un élève veut savoir s'il a la moyenne.
Pour cela, il dispose de ses notes représentées par les variables a, b, c et d.
1) Compléter l'algorithme afin qu'il calcule sa moyenne m et affiche si l'élève a la moyenne ou non.

Algorithme
a ←
b ←
c ←
d ←
m ←
Si
Alors afficher "l'élève a la moyenne"
Sinon

2) Traduire votre algorithme en Python

Python

3) Tester votre algorithme avec les notes suivantes.

1ère note	2ème note	3ème note	4ème note	L'élève a la moyenne?
7	8	12	14	
9,5	10	11	8	
6,5	12	13	14	

Exercice D6 :

Soit d la droite d'équation $y = 4x + 5$.
1) Ecrire un algorithme qui teste si un point dont on connaît les coordonnées appartient à la droite D.

Algorithme

Python

2) Tester votre algorithme avec les points suivants:

Point M	(0; 5)	(2; -7)	(-2; -3)	(-3; -2)	($\frac{11}{4}$; 16)
M ∈ D ?					

3) Modifier l'algorithme pour que l'on puisse, en plus, faire varier la droite (en y entrant le coefficient directeur et l'ordonnée à l'origine)

Algorithme

Python

Exercice D7 :

Un cinéma propose deux tarifs : avec ou sans abonnement.
 Sans abonnement, la place de cinéma coûte 10,50€
 L'abonnement pour un an coûte 50€ et la place revient alors à 6,30€.
 On souhaite concevoir un programme qui affiche directement si l'abonnement est plus avantageux pour un nombre N de places achetées.

1) Ecrire un algorithme permettant de répondre au problème et le traduire en Python.

Algorithme

Python

2) Tester votre algorithme avec les valeurs suivantes.

Nombre de films	2	7	10	15	25
Abonnement avantageux ?					

3) A l'aide d'une inéquation, déterminer le nombre de films à partir duquel l'abonnement est plus avantageux ?

Exercice D8 :

Une petite société implantée près d'une université propose des services de reprographie. La photocopie d'un document de format A4 (21,7 cm × 29,7 cm) coûte 8 centimes.
 Souhaitant fidéliser les étudiants et leur permettre d'obtenir des prix plus intéressants, le responsable propose, à la première visite de l'étudiant, l'abonnement annuel suivant: l'étudiant paie 20€ pour réaliser un nombre maximum de 500 photocopies, puis 6 centimes pour chaque photocopie au delà de la 500ème.
 Le responsable souhaite concevoir un programme qui affiche directement le prix P à payer par l'étudiant lors de sa première visite pour un nombre N de copies saisies.

Il devra commencer par indiquer si l'étudiant est abonné ou non.

On note T la variable tarif :

- pour le tarif 1(sans abonnement), on a T = 1
- pour le tarif 2 (avec abonnement), T = 2.

1) Ecrire un algorithme permettant de répondre au problème et le traduire en Python.

Algorithme

Python

2) Tester votre algorithme avec les valeurs suivantes.

Abonné?	Oui	Non	Oui	Non	Oui
Nombre de copies	250	250	625	625	800
Prix					

E - Boucles finies : Le « Pour »

Définition 9 : En algorithmique, on peut être amené à répéter un bloc d'instructions plusieurs fois.

La boucle "Pour" permet ceci.

Langage naturel	Python
Pour I allant de 1 à N Faire instructions FinPour	for I in range (1, $N+1$) : instructions

Ainsi ici, on répétera N fois les instructions contenues entre le "Faire" et le "FinPour".

Remarques :

- I peut être remplacé par n'importe quelle lettre.
- I peut commencer à une autre valeur que 1.
- On peut utiliser I dans le bloc d'instructions.
- Attention, en langage python, à la fonction range il ne faut pas oublier de « rajouter 1 ».
- Il ne faut pas oublier les deux-points à la fin.
- L'indentation (décalage) est obligatoire et se fait automatiquement sous Python.
- On supprime l'indentation pour sortir de l'instruction conditionnelle et poursuivre l'algorithme.
- De manière générale, l'instruction :
"Pour k variant de d à a " se traduit en Python par :
"for k in range($d, a+1$):"

Exercice E1 :

Un village compte aujourd'hui 2300 habitants.

Le village étant en pleine croissance, sa population augmente chaque année de 150 habitants.

On souhaite élaborer un algorithme donnant le nombre d'habitants de ce village dans n années.

On va donc devoir répéter n fois l'instruction "ajouter 150 à la population" : ce calcul avec des itérations se traduit avec une boucle bornée (boucle "pour")

Algorithme	Explications
$N \leftarrow ?$	- On demande le nombre d'années. Ce nombre est stocké dans la variable N .
$P \leftarrow 2300$	- On initialise P à 2300 (la population P de départ est à 2300)
Pour k variant de 1 à n	- k est un compteur qui part de 1, augmente de un en un et s'arrête à n . Il sert à compter le nombre de fois où l'on répète l'action.
$P \leftarrow P + 150$	- Action répétée n fois (on ajoute 150 à la population P pour obtenir la nouvelle valeur de P .)
Fin Pour	- Fin de la boucle quand k atteint la valeur n
Afficher P	- affichage du résultat

1) Faire fonctionner l'algorithme et compléter le tableau suivant pour $n = 4$.

Etape	Avant la boucle	Etape 1	Etape 2	Etape 3	Etape 4
k		1	2		
P	2300	2450			

Exercice E2 : Compléter le tableau suivant.

Algorithme	Python	Valeurs prises par k
Pour k allant de 2 à 6	for k in range(2,7)	2 ; 3 ; 4 ; 5 ; 6
Pour k allant de 5 à 10		
	for k in range(1,9)	
	for k in range(3,8)	
Pour k allant de 0 à 5		
	for k in range(7)	
Pour k allant de 1 à 3		

Exercice E3 :

Mme Econome effectue un unique virement de 20000 euros sur un livret d'épargne. Le capital est augmenté de 100 euros par an par le versement d'intérêts.

Compléter l'algorithme suivant afin qu'il affiche le capital disponible sur le livret dans 15 ans, puis le traduire en Python.

Algorithme	Python
$C \leftarrow \dots\dots\dots$	
Pour k variant de $\dots\dots\dots$	
$C \leftarrow \dots\dots\dots$	
Fin Pour	
Afficher $\dots\dots\dots$	

Quelle est la somme disponible au bout de 15ans ?

Exercice E4 :

Mme Econome effectue un virement de 3000 euros sur un livret d'épargne. Chaque année, le capital est augmenté de 3% (par le versement d'intérêts)

Compléter l'algorithme suivant afin qu'il affiche le capital disponible sur le livret dans 5 ans, puis le traduire en Python.

Algorithme	Python
$C \leftarrow \dots\dots\dots$	
Pour k variant de $\dots\dots\dots$	
$C \leftarrow \dots\dots\dots$	
Fin Pour	
Afficher $\dots\dots\dots$	

Quelle est la somme disponible au bout de 5ans ?

Exercice E5 :

En 2016, les rejets polluants d'un groupe industriel sont évalués à 5000 tonnes. Le groupe est contraint de réduire ses rejets polluants de 8% chaque année, jusqu'à ce que ceux-ci ne dépassent pas 2000 tonnes annuelles. On suppose que ce groupe respecte ce plan de réduction.

1) Par quelle valeur est multipliée la quantité de rejets polluants chaque année ?

2) Compléter l'algorithme afin qu'il affiche la quantité de rejets polluants du groupe au bout de N années .

Algorithme	Python
N ← ? R ← Pour k variant de R ← Fin Pour Afficher	

3) Faire tourner l'algorithme pour les valeurs de N suivantes :

N	2	6	8	11
Valeur en sortie				

Exercice E6 :

En prévision d'une course de vélo, Fanny suit le programme d'entraînement suivant sur 12 samedis : elle parcourt 25 km le premier samedi; puis elle augmente chaque semaine de 11km la distance parcourue.

1) Calculer la distance D parcourue le deuxième samedi puis la distance totale T parcourue au bout de deux samedis d'entraînement.

2) Créer un algorithme qui permet d'afficher la distance totale T parcourue au bout des 12 semaines d'entraînement

Algorithme
D ← T ← Pour k variant de D ← T ← Fin Pour Afficher

Python

Exercice E7 :

Pour commencer, nous allons nous intéresser au lancer d'une pièce en considérant que "Pile" correspondra au chiffre 1 et "Face" au chiffre 0.

Ainsi, on cherche à tirer aléatoirement un 0 ou un 1.

1) Indiquer ci-dessous la commande utilisée sur PYTHON pour obtenir ce nombre aléatoire :

2) Faire 10 simulations et compter le nombre de "Pile" obtenus :

3) On cherche à faire de même pour 500 simulations. Compléter l'algorithme ci-dessous qui permet de simuler 500 lancers de pièce et de compter le nombre de "Pile".

```
C ← .....  
Pour .....allant de .....à ..... Faire  
    P ← .....  
    Si P ..... Alors  
        .....← .....  
    FinSi  
Fin Pour  
Afficher .....
```

4) Modifier l'algorithme précédent pour simuler N lancers de pièce, où N est entré par l'utilisateur et le traduire en python.

Python

5) Le tester pour N = 5000. Combien de "Pile" sont obtenus? Quelle est la fréquence d'apparition du "Pile" pour cette simulation ?

F - Boucles non bornées : Le « Tant que »

Définition 10 : En algorithmique, on peut être amené à répéter un bloc d'instructions tant qu'une condition reste vérifiée.

On ne connaît pas à l'avance le nombre d'itérations mais uniquement la cause de poursuite (et donc par négation, la cause d'arrêt).

La boucle "Tant Que" permet ceci. Si la condition qui suit le "Tant Que" est vérifiée, alors le programme exécute toutes les instructions du bloc qui suit, sinon le bloc est entièrement ignoré.

Langage naturel	Python
Tant que <i>condition</i> Faire <i>instructions</i> FinTantque	while <i>condition</i> : <i>instructions</i>

Remarques :

- Il ne faut pas oublier les deux-points à la fin.
- L'indentation (décalage) est obligatoire et se fait automatiquement sous Python.
- On supprime l'indentation pour sortir de l'instruction conditionnelle et poursuivre l'algorithme.

Exercice F1 :

Une balle lâchée d'une hauteur de 90cm rebondit chaque fois qu'elle touche le sol aux $\frac{3}{5}$ de la hauteur du rebond précédent.

On veut créer un algorithme donnant le nombre de rebonds au bout duquel la hauteur atteinte par la balle est inférieure à 1cm. On va définir la variable H pour désigner la hauteur du rebond en cm. On va devoir répéter l'instruction "H prend la valeur $\frac{3}{5} \times H$ " sans savoir à l'avance le nombre de ces répétitions.

On va alors tester si la variable H est supérieure ou égale à 1cm pour savoir si on doit réaliser une fois de plus l'instruction "H prend la valeur $\frac{3}{5} \times H$ ".

On introduit aussi une variable N qui comptera le nombre de rebonds effectués par la balle.

Algorithme	Explications
N ← 0	- On initialise le nombre de rebonds à 0.
H ← 90	- On initialise H à 90 (c'est la hauteur de départ)
Tant que H ≥ 1	- On répète l'action tant que la condition est vérifiée (tant que la hauteur de la balle est plus grande que 1cm ; on arrête quand la hauteur est inférieure à 1cm).
H ← H × $\frac{3}{5}$	- calcul de la nouvelle hauteur atteinte par la balle
N ← N + 1	- On augmente le nombre de rebond de 1
Fin Tant que	- affichage du résultat
Afficher N	

1) Faire fonctionner l'algorithme et compléter le tableau suivant :

Etapas	Valeur de N	Valeur de H	Condition vérifiée
Avant la boucle	0	90	oui
N°1			

Valeur en sortie :

Interprétation :

2) Voici l'algorithme en Python :

```
1 N=0
2 C=90
3 while C>=1:
4     N=N+1
5     C=C*3/5
6 print("le nombre de rebonds nécessaires est",N)
```

Exercice F2 :

Mme Econome effectue un virement de 3000 euros sur un livret d'épargne en 2018.

Chaque année, le capital est augmenté de 3% (par le versement d'intérêts).

1) Quelle opération traduit l'augmentation de 3%?

2) Compléter l'algorithme suivant afin qu'il affiche l'année à partir de laquelle le capital disponible sur le livret sera supérieur à 3500€, puis le traduire en Python.

Algorithme	Python
C ← 3000	
A ← 2018	
Tant que	
C ←	
A ←	
Fin Tant que	
Afficher	

3) Quelle est la valeur affichée en sortie ?

Exercice F3 :

Dans une petite ville, on constate depuis quelques années une hausse annuelle de 5% du prix des loyers. Cette année, le prix moyen de location du mètre carré est 8€.

On souhaite déterminer dans combien d'années le prix moyen de location du mètre carré dépassera 12€.

1) Quelle opération traduit la hausse de 5% ?

2) Compléter l'algorithme ci-dessous afin que la variable N donne le nombre d'années cherché et le traduire en Python.

Algorithme	Python

3) Quelle est la valeur affichée en sortie ?

Exercice F4 :

Le 1er janvier 2017, Lucas reçu 90€ d'étérennes, puis chaque année, celles-ci augmentent de 5€. Il décide de ne pas dépenser cet argent avant de disposer de 700€.

1) Quel montant M recevra Lucas en 2018 ? en 2019 ?

2) De quelle somme totale T dispose-t-il en 2018 ? en 2019 ?

3) Compléter l'algorithme ci-dessous afin qu'il affiche l'année à partir de laquelle Lucas disposera de 700€.

Algorithme	Python
M ← 90	
T ← 90	
A ← 2017	
Tant que	
M ←	
T ←	
A ←	
Fin tant que	
Afficher A	

4) Quelle est l'année affichée en sortie ?

Exercice F5 :

Une entreprise de forage creuse des puits dans le désert afin d'atteindre la nappe d'eau phréatique. Cette entreprise facture le premier mètre 100€, le second mètre 140€ et ainsi de suite en augmentant le prix de chaque nouveau mètre creusé de 40€.

Une organisation humanitaire dispose de 4000€ pour faire creuser un puits.

On souhaite créer un algorithme qui permet de connaître la profondeur maximale du puits que peut financer cette organisation.

1) Compléter l'algorithme suivant et donner sa traduction en Python.

Algorithme	Python
M ← 100	
T ← 100	
N ← 1	
Tant que	
M ←	
T ← T + M	
N ←	
Fin tant que	
Afficher	

2) Modifier l'algorithme ci-dessus afin qu'il demande le montant disponible pour les travaux et affiche la profondeur du puits creusé avec cette somme.

Algorithme	Python

3) Tester l'algorithme ci-dessus pour les montants suivants :

Montant (en €)	5000	7000	10 000
Profondeur			

Exercice F6 :

Des chercheurs étudient l'évolution de la population de truites dans un lac de montagne. Ils constatent une diminution annuelle moyenne de 20% de cette population.

Afin de sauvegarder l'espèce, ils décident d'introduire chaque année 50 nouvelles truites.

On estime la population initiale du lac à 100 truites.

On souhaite savoir au bout de combien d'années la population dépassera les 230 individus.

1) Compléter l'algorithme suivant et donner sa traduction en Python.

Algorithme	Python
$T \leftarrow 100$ $N \leftarrow 0$ Tant que $T \leftarrow \dots\dots\dots$ $N \leftarrow \dots\dots\dots$ Fin tant que Afficher	

2) Quelle est l'année cherchée ?

Exercice F7 :

En 2016, les rejets polluants d'un groupe industriel sont évalués à 5000 tonnes. Le groupe est contraint de réduire ses rejets polluants de 8% chaque année, jusqu'à ce que ceux-ci ne dépassent pas 2000 tonnes annuelles. On suppose que ce groupe respecte ce plan de réduction.

1) Par quelle valeur est multipliée chaque année la quantité de rejets polluants?

2) Ecrire un algorithme donnant le nombre d'années nécessaires pour que les rejets ne dépassent pas 2000 tonnes annuelles.

Algorithme	Python

Programmation en Python, instructions de base

Lorsqu'on ouvre Python :

- la fenêtre à droite s'appelle **l'éditeur**,
c'est dans cette fenêtre qu'on écrit les programmes

- la fenêtre en bas s'appelle **la console**,
c'est dans cette fenêtre qu'on entre les instructions
qui vont faire tourner le programme,
et c'est là que les résultats s'affichent.

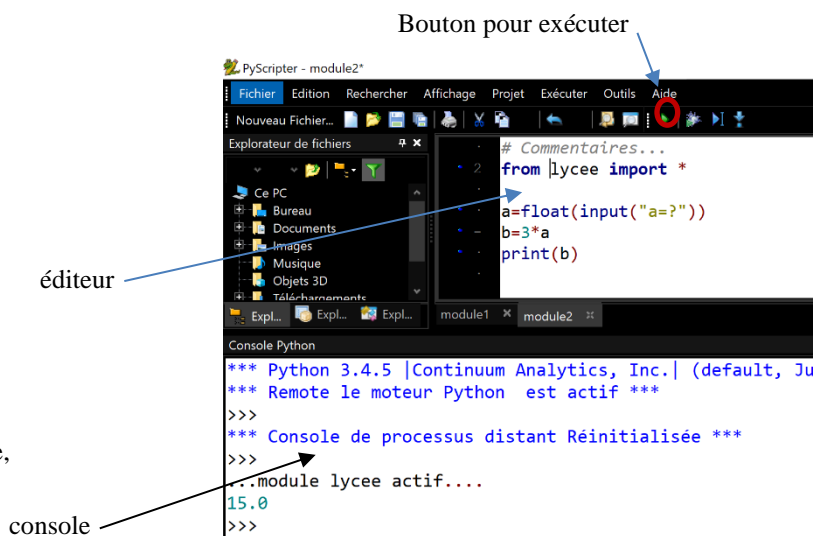
- le petit triangle vert dans la barre en haut
sert à exécuter le programme.

Généralités pour taper un programme :

- On peut écrire des commentaires, précédés de #.

- On va à la ligne après chaque instruction.

On peut aussi taper plusieurs instructions sur la même ligne,
en les séparant par un point-virgule.



Langage naturel	Python
$X \leftarrow ?$ (Saisir X ou Lire X)	<code>X=int(input("X="))</code> si X est un entier <code>X=float(input("X="))</code> si X est un flottant <code>X=input("X=")</code> si X est une chaîne de caractères
$X \leftarrow b$ (Affecter à X la valeur b)	<code>X=b</code>
Afficher X Afficher : « texte » Afficher : la réponse est X	<code>print(X)</code> <code>print("texte")</code> <code>print("la réponse est", X)</code>
Définir une fonction FONC dépendant des paramètres a, b et c pour obtenir un résultat R	<code>def FONC(a,b,c) :</code> ← ne pas oublier les deux-points Instructions R=... <code>return(R)</code>
Opérations :	* pour une multiplication / pour une division ** pour les puissances (ex : $2^{**}3$ signifie 2^3) // pour le quotient entier (ex : $7//3$ donne 2) % pour le reste de la division entière (ex : $7\%3$ donne 1)
Outils supplémentaires avec la bibliothèque lycée (écrire from lycée import * en début de programme ou bien ouvrir un nouveau fichier en choisissant directement le modèle « lycée », et cette instruction s'écrit toute seule).	• <code>pi</code> : π • <code>sqrt(A)</code> : la racine carrée du nombre A. • <code>uniform(a,b)</code> : nombre décimal aléatoire compris entre a et b. • <code>randint(a,b)</code> : nombre entier aléatoire compris entre a et b.
Si <i>condition</i> alors <i>instructions1</i> sinon <i>instructions2</i> Fin Si (ligne facultative si on respecte l'indentation)	<code>if condition :</code> ← ne pas oublier les deux-points <i>instructions1</i> <code>else :</code> ← ne pas oublier les deux-points <i>instructions2</i>
Pour tester si : $A \leq B$; $A \geq B$; $A = B$ et $A \neq B$	<code>A<=B ; A>=B ; A==B et A !=B</code>
Pour k allant de 1 à n : <i>instructions</i> Fin Pour (ligne facultative si on respecte l'indentation)	<code>for k in range(1,n+1):</code> ← ne pas oublier les deux-points <i>instructions</i>
Si on démarre à 0 : Pour k allant de 0 à n	<code>for k in range(0,n+1):</code> ou bien <code>for k in range(n+1) :</code>
Tant que <i>condition</i> : <i>instructions</i> Fin Tant que (ligne facultative si on respecte l'indentation)	<code>while condition :</code> ← ne pas oublier les deux-points <i>instructions</i>