



Algorithmes



Déclaration de variables

	Algorithme Pseudo-code	Code Python
Déclarer une variable a du type entier	a ← 4	a=4
Déclarer une variable b du type décimal	b ← 3.2	b=3.2
Déclarer une variable c du type chaîne de caractères.	c ← 'hello'	c="hello"

Compléments sur les variables:

<https://www.youtube.com/watch?v=nbecRjzVzNg&list=PLrSOXFDHBtfHg8fWBd7sKPxEmahwyVBkC&index=3>

Affichage

	Algorithme Pseudo-code	Code Python
Afficher le texte « Au revoir » avec un retour à la ligne.	affiche('Au revoir')	print("au revoir")
Afficher le texte « Résultat » suivi du contenu de la variable a.	affiche('résultat', a)	print("résultat",a)

Retrouver une valeur dans un tableau

A l'aide de la déclaration suivante :

tab=[3,5,9,10,14,17]

De combien de valeurs contient le tableau ?	6
Quel est l'indice min et l'indice max	3;17
Quel est l'instruction python qui permet d'afficher la valeur de l'indice 4	print(tab[4])
Quel est la valeur de l'indice 4	

La condition SI

Donner le résultat des programmes :

Algorithme Pseudo-code	Code python	Commentaires
a ← 15 si a > 15 alors affiche('bien') fin si	a = 15 if a > 15 : print('bien')	rien
a ← 15 si a == 15 alors affiche('bien') fin si	a = 15 if a == 15 : print('bien')	bien
a ← 7 si a > 15 alors affiche('bien') fin si Affiche('ok')	a = 7 if a > 15 : print('bien') print('ok')	ok
a ← 7 si a > 15 alors affiche('bien') affiche('ok') fin si	a = 7 if a > 15 : print('bien') print('ok')	rien
a ← 7 si a > 15 alors affiche('bien') affiche('ok') sinon affiche('non') fin si	a = 7 if a > 15 : print('bien') print('ok') else : print('non')	non
a ← 7 si a > 15 alors affiche('bien') affiche('ok') sinon affiche('non') affiche('inférieur') fin si	a = 7 if a > 15 : print('bien') print('ok') else : print('non') print('inférieur')	non inférieur

Quel est la différence entre = et == ?

Un = → **pour une affectation.** Exemple : a=3

Deux == → **pour un test dans une condition.**

Exemple : if a==3:
 print('la valeur de a est 3')

Complements sur les conditions:

https://www.youtube.com/watch?v=T_AsJ0nleh0&list=PLrSOXFDHBtfHg8fWBd7sKPxEmahwyVBkC&index=5

Opérateurs

Opération	Algorithme Pseudo-code	Code python	
Addition	$a \leftarrow a+5$	$a=a+5$	$a=b+9$
Incrémentation	$a \leftarrow a+1$	$a=a+1$	ou $a+=1$
Décrémentation	$a \leftarrow a-1$	$a=a-1$	ou $a-=1$
Soustraction	$b \leftarrow a-5$	$b=a-5$	
Multiplication	$c \leftarrow a*b$	$c=a*b$	
Puissance	$c \leftarrow a^2$	$C=a**2$	
Division entière	$a=43 // 5$ $c=a$ division entière par 5	Valeur de c :	8
Modulo (Reste de la division entière)	$a=43 \% 5$ $c=a$ modulo 5	Valeur de c :	3

La condition SI avec opérateurs logiques booléens

Donner le résultat des programmes :

Algorithme Pseudo-code	Code python	Commentaires
$a \leftarrow 2$ $b \leftarrow 3$ si $a=2$ et $b=3$ alors affiche('ok') sinon affiche('non') fin si	$a = 2$ $b = 3$ if $a == 2$ and $b == 3$: print('ok') else: print('non')	ok
$a \leftarrow 2$ $b \leftarrow 3$ si $a=2$ ou $b=3$ alors affiche('ok') sinon affiche('non') fin si	$a = 2$ $b = 3$ if $a == 2$ or $b == 3$: print('ok') else: print('non')	ok

Donner les tables de vérité du ou, et, non

a	b	S(ou)
False	False	
False	True	
True	False	
True	True	

a	b	S(et)
False	False	
False	True	
True	False	
True	True	

a	S(non)
False	
True	

La boucle Pour

Algorithme Pseudo-code	Code python	Résultat affiché sur la console
Pour n de 0 à 4 par pas de 1 Affiche(n) fin pour	for n in range(0,5): print(n,end=',')	1,2,3,4
Pour n de 0 à 10 par pas de 2 Affiche(n) fin pour	for n in range(0,11,2): print(n,end=',')	0,2,4,6,8,10
Pour n de 6 à 0 par pas de -1 Affiche(n) fin pour	for n in range(6,-1,-1): print(n,end=',')	6,5,4,3,2,1,0

Algorithme Pseudo-code	Résultat affiché sur la console		
	n	n*2	n//2
pour n de 1 à 3 par pas de 1 affiche(n) affiche(n*2) affiche(n//2) fin pour	1	2	0,5
	2	4	1
	3	6	1,5
			1er passage
			2ème passage
			3ème passage

<https://www.youtube.com/watch?v=BrknhzrHm8w>

La boucle tant que

Algorithme Pseudo-code	Code python	Résultat affiché sur la console
n ← 0 tant que n < 5 Affiche(n) n ← n + 1 fin tant que	n = 0 while n < 5: print(n,end=',') n = n + 1	0,1,2,3,5
n ← 6 tant que n ≥ 0 Affiche(n) n ← n - 1 fin tant que	n = 6 while n ≥ 0: print(n,end=',') n = n - 1	6,5,4,3,2,1,0

Complements sur les boucles:

<https://www.youtube.com/watch?v=excGUISppC4&list=PLrSOXFDHBtfHg8fWBd7sKPxEmahwyVBkC&index=6>