

TP2 : Premier pas avec les structures de base en python

Expressions et opérateurs

Affectation	Opérateurs numériques	Opérateurs alphanumériques	Opérateurs relationnels	Opérateurs logiques
Algorithmique				
←	+, -, *, ./, ^, //, %	+	>, <, =, ≤, ≥, !=	Et, Ou, Non
Programmation Python				
=	+, -, *, /, **, //, %	+	>, <, ==, <=, >=, !=	and, or, not

Exercice :

Écrire le programme python qui permet de lire :

- Le montant hors taxes d'un article : M_HT (un réel)
- Le taux de TVA : TVA (un réel)
- Le nombre d'articles : Nb_Articles (un entier)

Puis, le montant total toutes taxes comprises Total_TTC sachant que : $\text{Total_TTC} = \text{Nb_articles} * \text{M_HT} * (1 + \text{TVA})$

Structures sélectives

Structure sélective simple	Structure alternative	Structure imbriquée	Structure à choix multiple
if Conditions: Instructions	if Conditions: Instructions1 else: Instructions2	if Conditions1: Instructions1 elif Conditions2: Instructions2 else: Instructions3	match choix: case Choix1 : Instructions1 case Choix2 : Instructions2 case Choix3 : Instructions3 case _ : Instructions

Exercices :

1- Traduire les algorithmes ci-dessous en langage python, exécuter le programme et observer les résultats.

Algorithme Moyenne_decision; moyenne : REEL ; DEBUT AFFCIHER ("Votre moyenne :"); Lire (moyenne); SI moyenne >= 10 ALORS AFFCIHER ("admis"); SINON AFFCIHER ("non admis"); FINSI FIN.	ALGORITHME Nature_entier; N : ENTIER ; DEBUT AFFCIHER ("Donner un entier:"); LIRE (N); SI N > 0 ALORS AFFCIHER ("N : positif"); SINON SI N < 0 ALORS AFFCIHER("N : négatif"); SINON AFFCIHER("N : Nul"); FINSI FINSI FIN.	ALGORITHME chiffres_en_lettres; N : ENTIER ; DEBUT AFFCIHER ("Donner un caractère entre 1 et 5:"); LIRE (N); SELON N 1 : AFFCIHER ("Un"); 2 : AFFCIHER ("Deux"); 3 : AFFCIHER ("Trois"); 4 : AFFCIHER ("Quatre"); 5 : AFFCIHER ("Cinq"); SINON : AFFCIHER ("Chiffre non supporté"); FINSELON FIN.
--	--	--

2- Ecrire un programme python permettant de résoudre une équation du second degré : $ax^2+bx+c=0$.

Structures itératives

La structure Pour	La structure Tant que	La structure FAIRE ... TANT QUE
for i in range(valeur_initiale, max, pas): instructions	While conditions : instructions	while True: Instruction if condition: break

Exercices :

- Traduire les algorithmes ci-dessous en langage python, exécuter le programme et observer les résultats.

ALGORITHME Somme_N_Premiers

i, N, S : ENTIER

DEBUT

AFFICHER("Entrer un entier")

LIRE(N)

Somme \leftarrow 0

POUR (i \leftarrow 1, i \leq N, i \leftarrow i + 1)

S \leftarrow S + i

FINPOUR

AFFICHER("La somme des", N, " nombres entiers est :", S)

FIN

ALGORITHM Jeu_Devinette;

Constante nombre_secret = 7

N: ENTIER;

Début

FAIRE

AFFICHER ("Entrez un nombre : ")

LIRE(N);

TANTQUE (N= nombre_secret)

AFFICHER ("nombre secret trouvé.")

FIN.

- Écrire un programme python qui calcule le factoriel d'un nombre entier donné par l'utilisateur.
- Écrire un programme python pour trouver le plus grand diviseur de N > 1 autre que lui-même.
- Ecrire d'un programme demandant à l'utilisateur de saisir un nombre entier positif et qui permet de le convertir en binaire.
- Menu d'Options avec match et Boucles**
 - Demande à l'utilisateur de saisir un nombre entier positif.** Gérer les cas d'erreur, par exemple, si l'utilisateur saisit un nombre négatif ou une entrée non valide (utiliser la boucle **FAIRE ... TANTQUE**)
 - Affichage du Menu :** affichez un menu offrant au moins quatre actions différentes plus une option pour quitter , et demandez à l'utilisateur de choisir une option.
 - Vérifier et afficher si le nombre est un nombre premier
 - Tester si un entier donné en entrée est pair ou impair.
 - Convertir un nombre décimal non négatif en base b.
 - Calculer la factorielle d'un entier non négatif donnée en entrée.
 - Quitter le menu.

NB : Pour le traitement du Choix, Utilisez **match** pour exécuter un bloc de code spécifique selon l'option choisie.
- Boucle Continue :** Le programme doit continuer à exécuter, affichant le menu après chaque action, jusqu'à ce que l'utilisateur choisisse de quitter le programme.(utiliser la boucle **FAIRE ... TANTQUE**)

