

TD 1 : Instructions de base / Structures conditionnelles

Exercice 1 :

1. Qu'est-ce qu'un algorithme ?
2. Quelle est la différence entre Algorithme et Programme ?
3. Donner quelques exemples des langages de programmation.
4. Quel est le résultat de l'étape d'analyser un problème.
5. A quoi sert l'instruction d'affectation ?
6. Rappeler le rôle des instructions d'entrée/sortie.

Exercice 2 :

Quelles seront les valeurs des variables a, b, c et d après exécution des instructions suivantes :

```
a ← 3  
b ← 7  
c ← a + b  
b ← 2  
a ← c - b  
c ← a * b  
d ← a mod c
```

Exercice 3 : Somme de deux nombres

Ecrire un algorithme qui lit deux nombres réels, puis calcule et affiche leur somme.

Exercice 4 : Aire et périmètre d'un rectangle

Écrire un algorithme en pseudo-code qui lit la longueur et la largeur d'un rectangle, puis calcule et affiche son aire et son périmètre.

Exercice 5 :

Ecrire un algorithme qui lit un entier s (supposé positif) représentant un nombre de secondes, et qui affiche ce nombre dans le format suivant : x H y M z S, tel que x est le nombre d'heures, y le nombre de minutes et z le nombre de seconds restants.

Exemple :

Pour s=4000
On aura le résultat suivant : 1 H 6 M 40 S

Exercice 6 :

Ecrire un algorithme qui détermine si un nombre entré par l'utilisateur est pair ou impair.

Exercice 7 :

Ecrire un **algorithme en pseudo** code puis en **Python** qui fait la résolution des équations du premier degré de la forme :
 $aX + b = 0$

Exercice 8 :

Ecrire un algorithme qui affiche la mention associée à une note (sur 20) entré par l'utilisateur :

- « Échec » si la note < 10
- « Passable » si la note ≥ 10 et < 12
- « Assez Bien » si la note ≥ 12 et < 14
- « Bien » si la note ≥ 14 et < 16
- « Très bien » si la note ≥ 16