

II - Les instructions d'entrée / sortie :

Pour bien, fonctionner, un algorithme doit en général acquérir des données (entrées) sur un périphérique (Exemple : clavier, disque) et fournir en conséquence des résultats (sorties) sur un autre périphérique (écran, imprimante).

Pour réaliser ces tâches fondamentales en programmation, on a donc besoin de deux opérations distinctes :

Lire(V) : qui permet **d'affecter à la variable (V), la valeur** lue sur le périphérique d'entrée.

Ecrire(V) : qui permet de transférer la **valeur (V)** vers le périphérique de sortie.

2.1 - Remarque 1:

On peut avoir la même instruction "Ecrire" l'affichage d'un ou plusieurs messages et d'une ou plusieurs variables.

2.2 - Exemple :

Ecrire " La somme de ces deux nombres " , a , " et " , b , " est : " , S
Si a = 2 et b = 5, l'instruction ci-dessus s'exécutera ainsi :
La somme des deux nombres 2 et 3 est : 7

2.3 - Remarque 2:

L'identificateur (V) doit être une variable déclarée. Par conséquent les écritures suivantes n'ont pas de sens :

Lire(6) ;

Lire(A+B) ;

2.4 - Exemple :

VARIABLES A, B, SOMME : ENTIERS

Début

ECRIRE 'Entrez le premier nombre'

Lire A

ECRIRE 'Entrez le deuxième nombre'

Lire B

SOMME \leftarrow A + B

ECRIRE 'La somme de ces deux nombres est : '

ECRIRE SOMME

Fin

Exercices :

1. Quel résultat produit le programme suivant ?

VARIABLES Val, Double : ENTIERS

Début

Val \leftarrow 231

Double \leftarrow Val * 2

ECRIRE Val

ECRIRE Double

Fin

2. Ecrire un programme qui demande deux nombres entiers à l'utilisateur, puis qui calcule et affiche la somme de ces nombres.
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.
4. Ecrire un programme qui lit une valeur et qui nous calcule l'inverse de cette valeur.
5. Le surveillant général d'un établissement scolaire souhaite qu'on lui écrit un programme qui calcule, pour chaque élève, la moyenne des notes des cinq matières. Ces matières sont avec leur coefficient:

MATIERE	COEFFICIENT
Math	5
Physique	5
Français	4
Anglais	2
Histoire – Géographies	2

III - La structure alternative :

1 - Les conditions simples :

1.1 - Syntaxe :

Si condition **Alors**
 Instruction (ou suite d'instructions)
Finsi

1.2 - Exemple :

Si $x < 0$ **Alors**
 Ecrire " Valeur négatif "
Finsi

2 - L'instruction Si ... Alors ... Sinon ... Finsi :

2.1 - Syntaxe :

Si condition **Alors**
 Instruction1
Sinon
 Instruction2
Finsi

2.2 - Exemple :

```
Si  $x < 0$  Alors
    Ecrire " Valeur négatif "
Sinon Si  $x > 0$  Alors
    Ecrire " Valeur positif "
Finsi
```

```
Si  $x < 0$  Alors
    Ecrire " Valeur négatif "
Sinon
    Ecrire " Valeur positif "
Finsi
```

3 - L'instruction Selon :

3.1 – Syntaxe :

Selon variable

```
Cas valeur1
    Bloc1
Cas valeur2
    Bloc2
Cas valeur3
    Bloc3
```

Fin Selon

Exercices :

1. 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 le produit des deux nombres.
2. Ecrire un algorithme qui demande trois noms à l'utilisateur et l'informe ensuite s'ils sont rangés ou non dans l'ordre alphabétique.
3. Ecrire un algorithme qui demande un nombre à l'utilisateur, et l'informe ensuite si ce nombre est positif ou négatif (on inclut cette fois le traitement du cas où le nombre vaut zéro).
4. Ecrire un algorithme qui demande deux nombres à l'utilisateur et l'informe ensuite si le produit est négatif ou positif (on inclut cette fois le traitement du cas où le produit peut être nul). Attention toutefois, on ne doit pas calculer le produit !
5. Ecrire un algorithme qui demande l'âge d'un enfant à l'utilisateur. Ensuite, il l'informe de sa catégorie :
 - « Poussin » de 6 à 7 ans
 - « Pupille » de 8 à 9 ans
 - « Minime » de 10 à 11 ans
 - « Cadet » après 12 ans
6. a partir d'un montant lu, on détermine un montant net par application d'une remise de :
 - 1% si le montant est compris entre 2000 et 5000 DH (valeurs comprises)
 - 2 % si le montant est supérieur à 5000 DH.
7. Ecrire un algorithme permettant d'effectuer la somme, la différence, le produit et le rapport de deux nombres réels à partir d'un menu :

*****Menu*****

```

1 :-----Somme-----
2 :-----Différence-----
3 :-----Produit-----
4 :-----Division-----
5 :-----Quitter-----

```

Taper votre choix?

On doit retourner au menu, lorsqu'on finit une opération ou si on tape un nombre incorrect (un message s'affichera).

8. Saisir 2 entiers a et b, déterminer dans R les racines de l'équation : $ax + b = 0$

9. Saisir 3 entiers a, b et c, déterminer dans R les racines de l'équation : $aX^2 + bX + c = 0$

10. Donnez un algorithme qui lit sur l'entrée une valeur représentant une somme d'argent et qui calcule et affiche le nombre de billets de 200 DH, 100 DH, 50 DH et 20 DH, et les pièces de 10 DH, 5 DH, 2 DH, et 1 DH.

IV - Les structures répétitives :

1 - L'instruction Pour :

1.1 - Syntaxe :

POUR *compteur* = *val_initial* **A** *val_final* **PAS DE** *incrément*
 Instructions à répéter

FIN POUR

1.2 - Exemple :

Variable i Entier

Début

Pour i de 1 à 3 Faire

Ecrire « Hello World »

FinPour

Fin

Résultat :

Hello World
 Hello World
 Hello World

Exercices :

1. 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

2. 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 :

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

3. Ecrire un algorithme qui demande un nombre de départ, et qui calcule sa factorielle.

NB : la factorielle de 8, notée 8 ! vaut $1 \times 2 \times 3 \times 4 \times 5 \times 6 \times 7 \times 8$

4. Ecrire un algorithme qui demande successivement 20 nombres à l'utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces 20 nombres :

Entrez le nombre numéro 1 : 12

Entrez le nombre numéro 2 : 14

...

Entrez le nombre numéro 20 : 6

Le plus grand de ces nombres est : 14

Modifiez ensuite l'algorithme pour que le programme affiche de surcroît en quelle position avait été saisie ce nombre :

C'était le nombre numéro 2

5. Ecrire un algorithme qui :

- lit d'abord une valeur

- ensuite il va lire successivement 20 nombres.

- enfin il va déterminer combien de fois la première valeur a été saisie (sans compter la première saisie).

6. Ecrire un algorithme qui calcule et affiche la valeur de Y donnée par :

$$Y = N^{10} \quad \text{Si } N \geq 20$$

$$Y = N! \quad \text{Si } N < 20$$

N étant un entier positif connu.

7. Ecrire un algorithme pour tester si un nombre est parfait. Un nombre est dit parfait s'il est égal à la somme de ses diviseurs stricts.

Exemple : $28 = 1 + 2 + 4 + 7 + 14$.

8. Un nombre entier P est premier si ses seuls diviseurs sont 1 et P. Ecrire un algorithme qui permet de déterminer si un nombre est premier ou non.

2 - La structure TANT QUE :

2.1 - Syntaxe :

Tantque condition de continuation **Faire**
Instructions à répéter

FinTantque

2.2 - Exemple :

Variable ch chaîne de caractère

Début

ch = « oui »

Tantque ch = « oui » **Faire**

Ecrire « Hello World »

Ecrire « Voulez-vous continuer oui/non »

Lire ch
FinTantque
Ecrire « Fin »

Fin

Résultat :

Hello World
 Voulez-vous continuer oui/non : **Si** la réponse est égale oui le programme affiche
 "Hello World" **Si** la réponse est égale non le programme affiche Fin.

Exercices :

1. Ecrire un algorithme qui demande à l'utilisateur un nombre compris entre 1 et 3 jusqu'à ce que la réponse convienne.
2. 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.
3. Ecrire un algorithme qui demande un nombre de départ, et qui ensuite affiche les dix nombres suivants. Par exemple, si l'utilisateur entre le nombre 17, le programme affichera les nombres de 18 à 27.
4. Ecrire un algorithme qui demande successivement des nombres à l'utilisateur, et qui lui dise ensuite quel était le plus grand parmi ces nombres et quel était sa position. La saisie des nombres s'arrête lorsque l'utilisateur entre un zéro.
5. Lire la suite des prix (en DH entiers et terminée par zéro) des achats d'un client. Calculer la somme qu'il doit, lire la somme qu'il paye, et déterminer le reste à rendre.

3 - La structure REPETER :

3.1 - Syntaxe :

Répéter

Instruction à répéter

Jusqu'à condition

3.2 - Exemple :

Variable n Entier

Début

Répéter

Ecrire « Entrez un nombre : »

Lire n

Jusqu'à n = - 1

Ecrire « Fin »

Fin

Résultat :

Entrez un nombre : **Si** la réponse est égale un nombre différent à (-1) le programme affiche " Entrez un nombre : ". **Si** la réponse est égale (-1) le programme affiche "Fin".

Exercices :

1. Ecrire un algorithme qui demande successivement des nombres à l'utilisateur, et qui calcule le nombre de valeurs saisies. La saisie des nombres s'arrête lorsque l'utilisateur entre le caractère « n » ou « N ».
 2. Ecrire un algorithme qui demande successivement des nombres à l'utilisateur, et qui calcule leur moyenne. La saisie des nombres s'arrête lorsque l'utilisateur entre un zéro.
 3. Modifiez l'algorithme de l'exercice 1, de façon qu'il nous renseigne sur le nombre des valeurs positives et sur le nombre des valeurs négatives. Ne comptez pas les valeurs nuls.
 4. Ecrire un algorithme qui lit les caractères saisis par l'utilisateur. A la fin ce programme nous affichera la phrase saisie. La saisie des caractères s'arrête lorsqu'on tape point « . ». Pour l'utilisateur veut insérer un espace il lui suffit de taper sur 0. Par exemple si l'utilisateur tape successivement les caractères « b », « o », « n », « j », « o », « u », « r », « t », « o », « u », « s », « . », il nous affichera la chaîne « bonjour tous ».
- Mais si il tape « b », « o », « n », « j », « o », « u », « r », « 0 », « t », « o », « u », « s », « . », le programme affichera « bonjour tous ».

V - Les tableaux à une seule dimension :

Une donnée de type tableau est constituée d'un nombre de données élémentaires de même type.

5.1 - Syntaxe :

Variable Tableau nomTableau(Taille) en type

5.2 - Exemples :

Tableau Note(5) : Réel

Note (5) est un tableau qui contient vingt valeurs réelles.

Tableau nom(10), prénom(10) : Chaîne de caractère

Nom(10) et prénom(10) sont deux tableaux de 10 éléments de type chaîne.

Un tableau peut être représenté graphiquement par (exemple Note(5)) :

Note (1)	Note (2)	Note (5)
----------	----------	-------	-------	----------

Exercices :

1. Considérons les programmes suivants:

Tableau X (4) : Entier

DEBUT

X (1) \leftarrow 12

X (2) \leftarrow 5

X (3) \leftarrow 8

X (4) \leftarrow 20

FIN

Tableau voyelle (6) : Chaîne

DEBUT

Voyelle (1) \leftarrow « a »

Voyelle (2) \leftarrow « e »

Voyelle (3) \leftarrow « i »

Voyelle (4) \leftarrow « o »

Voyelle (5) \leftarrow « u »

Voyelle (6) \leftarrow « y »

FIN

Donner les représentations graphiques des tableaux X (4) et voyelle (6) après exécution de ces programmes.

2. Quel résultat fournira l'exécution de ce programme :

Variable i : Entier

Tableau C (6) : Entier

DEBUT

POUR i = 1 **A** 6

Lire C (i)

FIN POUR

POUR i = 1 **A** 6

 C (i) \leftarrow C (i) * C (i)

FIN POUR

POUR i = 1 **A** 6

Ecrire C (i)

FIN POUR

FIN

Si on saisit successivement les valeurs : 2 , 5 , 3 , 10 , 4 , 2.

3. Que fournira l'exécution de ce programme :

Tableau suite (8) : Entier

Variable i : Entier

DEBUT

Suite (1) \leftarrow 1

Suite (2) \leftarrow 1

POUR i = 3 **A** 8

 Suite (i) \leftarrow suite (i - 1) + suite (i - 2)

FIN POUR

POUR i = 1 **A** 8

Ecrire suite (i)

FIN POUR

FIN

4. Soit T un tableau de vingt éléments de types entiers. Ecrire le programme qui permet de calculer la somme des éléments de ce tableau.

5. Soit T un tableau de N entiers. Ecrire l'algorithme qui détermine le plus grand élément de ce tableau.