

# Exercice 1

Faite le déroulement des algorithmes suivants en donnant la valeur finale de chaque variable :

<p><b>Algorithme A1</b> Variables a,b: entier Début</p> <p>→ a ← 1 → b ← 2 → a ← a+3 → b ← a+b → b ← b-a</p> <p>Fin</p>	<p><b>Algorithme A2</b> Variables a,b,c: entier Début</p> <p>→ a ← 5 → b ← a+5 → c ← a+b → b ← c-a → a ← c*b</p> <p>Fin</p>	<p><b>Algorithme A3</b> Variables a,b,c: entier Début</p> <p>→ a ← 5 → b ← a+4 → a ← b+1 → b ← 2 → c ← a-b</p> <p>Fin</p>
---	---	---

a ~~1~~  
4

b ~~2~~  
~~6~~  
2

a ~~5~~    b ~~10~~    c 15  
45    5

a ~~5~~    b ~~9~~    c 8  
10    2

La valeur finale  
de a est 4

La valeur finale  
de b est 2

**NB :**  
— L'affectation attribue une nouvelle valeur et efface (érase) l'ancienne valeur de la variable.

## Exercice 2

Faite le déroulement des algorithmes suivants en donnant la valeur finale de chaque variable :

<p><b>Algorithme Alg1</b> Variables a,b,c,d: entier Début → a ← 10+2*5-1 b ← 10+2*(5-1) c ← (10+2)*5-1 d ← (10+2)*(5-1) Fin</p>	<p><b>Algorithme Alg2</b> Variables a,b: entier c :réel Début a ← 20/3 b ← 30 mod 7 c ← 20 div 3 Fin</p>	<p><b>Algorithme Alg3</b> Variables a,b,c: booléen Début a ← VRAI b ← FAUX → c ← a ET b → b ← c OU a → a ← NON(b ET c) Fin</p>
---	--	--

$$a \leftarrow 10 + (2 * 5) - 1 \text{ (soit 19)}$$

$$a \boxed{19}$$

$$b \leftarrow 10 + [2 * (5 - 1)] \text{ (soit 18)}$$

$$b \boxed{18}$$

$$c \leftarrow [(10 + 2) * 5] - 1 \text{ (soit 59)}$$

$$c \boxed{59}$$

$$d \leftarrow (10 + 2) * (5 - 1) \text{ (soit 48)}$$

$$d \boxed{48}$$

$$a \leftarrow 20/3$$

(a est entier, donc  
a ← entier(20/3)  
soit 6)

$$a \boxed{6}$$

$$b \leftarrow 30 \text{ mod } 7 \text{ (soit 2)}$$

$$b \boxed{2}$$

$$c \leftarrow 20 \text{ div } 3$$

(c est réel donc  
c ← réel(20 div 3)  
soit 6.0)

$$c \boxed{6.0}$$

$$a \boxed{\checkmark} \quad b \boxed{\cancel{F}} \quad c \boxed{F}$$

↓                    ↓

Absence d'instruction de sortie

### Exercice 3

Que produisent les algorithmes suivants ?

<p><b>Algorithme concat1</b>  Variables A, B, C: chaîne de Caractères</p> <p>Début  A ← "Alg"  B ← "32"  C ← A + B</p> <p>Fin</p>	<p><b>Algorithme concat2</b>  Variables A, C: chaîne de Caractères  B : entier</p> <p>Début  A ← "Alg"  B ← 32  C ← A + B</p> <p>Fin</p>	<p><b>Algorithme concat3</b>  Variables A, B: chaîne de Caractères  C : entier</p> <p>Début  A ← "Alg"  B ← "32"  C ← A + B</p> <p>Fin</p>
---	--	--

a "Alg"      b "32"

$C \leftarrow A + B$  (soit "Alg" + "32")

c "Alg32"

(Opération de concaténation et non pas d'addition)

$C \leftarrow \text{"Alg"} + 32$

chaîne      entier

L'instruction génère une erreur.

NB :

Il existe des langage de pointe qui adopte ce genre d'opération en considérant le code ASCII de l'entier (ici 32)

$C \leftarrow \text{"Alg"} + \text{"32"}$  (soit "Alg32")

Or c est déclaré entier.

$\Rightarrow$  L'instruct<sup>n</sup> génèrera une erreur.

### Exercice 4

Considérons les algorithmes ci-dessous. Quelles seront les valeurs des variables A et B l'exécution ?

N est un entier positif (  $N \neq 0$  et  $N \neq 1$  )

2m

<p><b>Algorithme A1</b>  <b>Variables</b> A,B,N : entier                            C : booléen</p> <p><b>Début</b>          A ← N mod N          B ← N mod (N div 2)          C ← A=B</p> <p><b>Fin</b></p>	<p><b>Algorithme A2</b>  <b>Variables</b> A,B,N : entier</p> <p><b>Début</b>          A ← N mod (N*N)          B ← (N*N) mod N</p> <p><b>Fin</b></p>	<p><b>Algorithme A3</b>  <b>Variables</b> A,B,N : entier</p> <p><b>Début</b>          A ← (2*(N+1)*(N+1)+1) mod 2          B ← (4*N*N+6) mod 2</p> <p><b>Fin</b></p>
--	--	--

$A \leftarrow N \bmod N$  (soit 0)

$$\begin{array}{r} N \\ 0 \overline{) N} \\ 0 \end{array}$$

$B \leftarrow N \bmod (N \text{ div } 2) = \begin{cases} 0 & \text{si } N \text{ est pair} \\ 1 & \text{si } N \text{ est impair} \end{cases}$

$C = \begin{cases} \text{vrai} & \text{si } A=B \\ \text{faux} & \text{si } A \neq B \end{cases}$

$$\begin{array}{r} N \\ N \overline{) N*N} \\ 0 \end{array}$$

$A \leftarrow N$

---


$$\begin{array}{r} N*N \\ 0 \overline{) N*N} \\ 0 \end{array}$$

$B \leftarrow 0$

$A \leftarrow (2*(N+1)*(N+1)+1) \bmod 2$

$A \leftarrow \underbrace{2(N+1)^2 + 1}_{\text{impair}} \bmod 2$

$\Rightarrow A \leftarrow 1$

---

$B \leftarrow (4N*N+6) \bmod 2$

$B \leftarrow (4N^2+6) \bmod 2$

$B \leftarrow \underbrace{(2(2N^2+3))}_{\text{pair}} \bmod 2$

$\Rightarrow B \leftarrow 0$

## Exercice 5

Citer et corriger les erreurs commises dans les deux algorithmes suivants :

```
Algorithme Erreur1
Constante R=3
Variables x : entier
Début
  Lire (R)
  x ← 1
  y ← R-x
  Ecrire ("Y=Y")
Fin
```

```
Algorithme Erreur2
Variables x,y : entier
Début
  Ecrire (Donner x, y)
  Lire (x,y)
  x ← y+1
  z ← z+x
  Ecrire ("x,z")
Fin
```

Variables  
x, y, R : entier

Début  
Lire (R)  
x ← 1  
y ← R-x  
Ecrire ("Y=", y)

Fin

Variables  
x, y, z : entier

Début  
Ecrire ("Donner x, y, z :")  
Lire (x, y, z)  
x ← y+1  
z ← z+x  
Ecrire (x, z)

Fin



## Exercice 7

Soit l'algorithme suivant :

```
Algorithme Exe7
Variables val, db, tr : entier
Début
  val ← 1000 ;
  db ← val * 2 ;
  tr ← val * 3 ;
  Ecrire (val) ;
  Ecrire (db) ;
  Ecrire (tr) ;
Fin
```

a) Quel résultat produira-t-il le déroulement de l'algorithme suivant ?

b) Proposer une simplification de cet algorithme en produisant le même résultat.

a)

val 1000    db 2000    tr 3000

le programme affichera les valeurs :

1000  
2000  
3000

Début

val ← 1000  
db ← val \* 2  
tr ← val \* 3

Ecrire (val, db, tr)

Fin

## Exercice 8

Supposons que l'on veut afficher à l'écran de l'utilisateur le message « Bonjour à tous ».

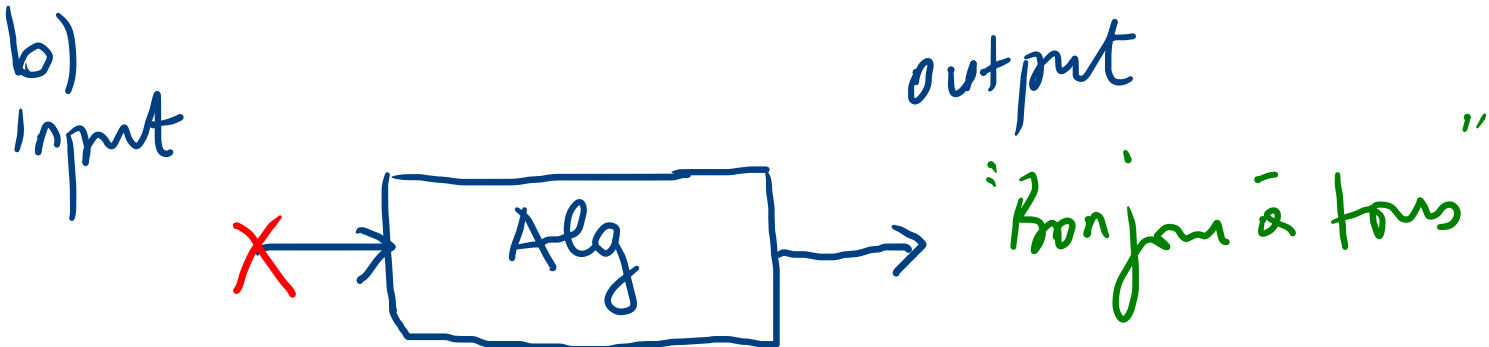
a) Essayer d'écrire l'algorithme correspondant.

b) Quelle est la sortie (output) de cet algorithme ? et l'entrée (input) ?

c) Quelles sont les instructions à utiliser pour manipuler les entrées (et les sorties) d'un algorithme ?

d) Reformuler votre algorithme pour qu'il y ait une entrée et une sortie.

a)  
Ecrire ("Bonjour à tous")



c)  
Les entrées : L'instruction de lecture  
Les sorties : ' ' d'écriture

d)

```
Var  
  nom: chaîne de caractère  
  
Debut  
  Ecrire ("Tape ton nom :")  
  Lire (nom)  
  Ecrire ("Bonjour", nom)  
  
Fin
```

## Exercice 9

- a) Ecrire un algorithme qui permet de lire un nombre (donné par l'utilisateur), puis il calcule et affiche son carré.
- b) Même question pour calculer et afficher le cube, le double, la moitié ensuite l'inverse de ce nombre.

a)

Variable  $n$ : entier

Début

Ecrire ("Tapez n: ")

Lire ( $n$ )

Ecrire ("carré = ",  $n * n$ )

Fin

{ sans déclaration de variables }  
{ supplémentaires }

b)

Var  $n, db, cub$ : entier

$moit, inv$ : réel

Début

Lire ( $n$ )

$db \leftarrow n^2$

$cub \leftarrow n^3$

$moit \leftarrow n/2$

$inv \leftarrow 1/n$

Ecrire ("double = ",  $db$ , "\n cube = ",

" moitié = ",  $moit$ , "\n inverse = ",  $inv$ )

Fin

{ Avec déclaration de variables }  
{ supplémentaires ( $db, cub, moit$  et  $inv$ ) }

NB

'\n' : Le caractère retour à la ligne ( $\leftarrow$ )

## Exercice 10

a) Ecrire un algorithme qui permet de lire les notes de trois matières, ensuite il calcule et affiche leur moyenne.

b) Modifier l'algorithme dans le cas où les matières ont des coefficients qui doivent être donnés avec les notes.

a)

```
Variable
note1, note2, note3 : réel
Début
  Ecrire ("Taper les 3 notes respectivement:")
  Lire (note1, note2, note3)
  Ecrire ("La moyenne =", (note1+note2+note3)/3)
Fin
```

b)

```
var
  n1, n2, n3, c1, c2, c3, moy : réel
Début
  Ecrire ("Taper les notes:")
  Lire (n1, n2, n3)
  Ecrire ("Taper les coefficients respectivement:")
  Lire (c1, c2, c3)
  moy ← (n1*c1 + n2*c2 + n3*c3) / (c1 + c2 + c3)
  Ecrire ("Moyenne =", moy)
Fin
```

## Exercice 11

a) Ecrire un algorithme qui permet de lire deux variables numériques a et b, de les afficher avant et après leur permutation.

Par exemple, avant : a=15 et b=6, après : a=6 et b=15.

b) Proposer un algorithme qui réalise la permutation de deux variables numériques sans avoir utiliser une troisième variable.

a)

```
var
  a, b, tmp : entier
Début
  Ecrire ("Tapez a et b :")
  Lire (a, b)
  Ecrire ("avant a =", a, " et b =", b)

  tmp ← a
  a ← b
  b ← tmp
  Ecrire ("après a =", a, " et b =", b)
Fin
```

b)

```
Variable
  a, b : entier
Début
  Lire (a, b)
  a ← a + b
  b ← a - b
  a ← a - b
  Ecrire (a, b)
Fin
```

Exemple :

a 15    b 6

----- ~~21~~ ----- 15

----- 6 -----

## Exercice 12

Ecrire un algorithme qui demande à l'utilisateur de saisir son nom et son prénom puis afficher « Bonjour » suivi de nom et de prénom de l'utilisateur.

Variable

nom, prenom : chaîne de caractères

Début

Ecrire ("Tapez votre nom :")

Lire (nom)

Ecrire ("Tapez votre prénom :")

Lire (prenom)

Ecrire ("Bonjour ", nom, " ", prenom)

Fin

' ' : caractère espace

### Exercice 13

Ecrire un algorithme 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.

Exemple d'une facture :

produit	P <sub>UHT</sub>	Qté	MHT
ordinateur	700,00	2	1400,00
Taux TVA			10 %
MTVA			140,00
TTC			1540,00

$$\begin{aligned}MHT &= P_{UHT} * Qté \\MTVA &= MHT * TTVA \\TTC &= MHT + MTVA \\&= MHT + MHT * TTVA \\TTC &= MHT (1 + TTVA)\end{aligned}$$

Variable

PHT, nbre, TTVA, TTC : réel

Début

Lire (PHT, nbre, TTVA)

TTC ← (PHT \* nbre) \* (1 + TTVA)

Ecrire ("TTC = ", TTC)

Fin

## Exercice 14

Ecrire l'algorithme permettant de calculer et d'afficher le salaire net d'un employé.

Sachant que :

- Le salaire net (SN) = Salaire brut - Valeur de l'impôt
- Salaire brut (SB) = (Salaire de base (SDB) + Prime des enfants) \* Taux de travail
- Taux de travail (TT) = Nombre de jours travaillés (NJ) / 26
- Prime des enfants (PE) = Prime d'un enfant (P1E) \* Nombre d'enfants (NE)
- Valeur de l'Impôt (VI) = Taux de l'Impôt (TI) \* Salaire Brut

Variable

SN, SB, SDB, TT, PE, P1E, VI, TI : réel  
NJ, NE : entier

Début

Ecrire ("Saisir salaire de base, Taux d'impôt respectivement :")

Lire (SDB, TI)

Ecrire ("Saisir la prime d'un enfant et le nbre d'enfant respectivement :")

Lire (P1E, NE)

Ecrire ("Saisir le nbre de jours travaillés :")

Lire (NJ)

TT ← NJ / 26

PE ← P1E \* NE

SB ← (SDB + PE) \* TT

VI ← TI \* SB

SN ← SB - VI

Ecrire ("Le salaire Net =", SN)

Fin