

---

# INFORMATIQUE 1 - SÉRIE DE TP N°03

---

## Sommaire

<b>Série de TP N°03 (Lire, Écrire &amp; Affectation).....</b>	<b>2</b>
<b>Solution.....</b>	<b>3</b>
<b>Exercice N°01 : Algorithmes → Programme.....</b>	<b>3</b>
1- Traduire l'algorithme en Programme PASCAL.....	3
2- Formatage de l'affichage des valeurs réelles.....	3
3- Dérouler le programme (ou l'algorithme) pour $R = 3$ .....	4
4- Compléter le schéma Entrées, Traitement et Sorties.....	4
5- Remarque (L'organigramme).....	5
<b>Exercice N°02 : Énoncé de problème → algorithme → Programme.....</b>	<b>5</b>
1 – Permuter entre les deux variables X et Y ?.....	5
2 – Permuter entre les trois variables X , Y et Z ?.....	6
3 – Calculer la division entre deux variables réelles a et b.....	7
4 – Calculer la division euclidienne entre deux entiers a et b.....	8
5 – Calculer la somme de a et b et le produit de b et c.....	9
6– Calculer la valeur absolue, le carré et la racine carrée d'un nombre.....	9

## TP INFORMATIQUE 1

### SÉRIE DE TP N°03 (LIRE, ÉCRIRE & AFFECTATION)

#### EXERCICE N°01 : ALGORITHMES → PROGRAMME

Soit l'algorithme suivant :

```

Algorithme Exo1;
Constantes
    Pi=3.14;
Variables
    R,P,S : réel;
Début
    {Entrées}
    Écrire('Donner la valeur du rayon R:');
    Lire(R);

    {Traitement}
    P ← 2*PI*R;
    S ← PI*R*R;

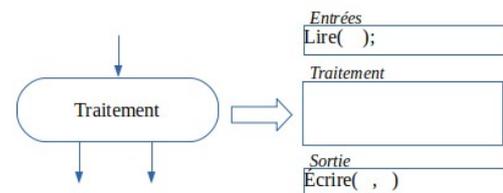
    {Sorties}
    Écrire('Périmètre = ', P);
    Écrire('Surface = ', S);
Fin.
    
```

#### Questions

- 1- Traduire l'algorithme en Programme PASCAL, puis compiler et exécuter le programme ?
- 2- Remplacer les deux dernières instructions par :  
 Écrire('Périmètre = ', P:0:3);  
 Écrire('Surface = ', S:10:2);  
 Puis exécuter le programme.  
 Qu'est-ce-que vous déduisez ?
- 3- Dérouler le programme pour R=3 ?

Instructions	Variables				Affichage
	V1	V2	...	Vn	
instruction_01					
instruction_02					
...					
...					

4- Compléter le schéma suivant :



#### EXERCICE N°02 : ÉNONCÉ DE PROBLÈME → ALGORITHME → PROGRAMME

Écrire un algorithme, puis traduit le en programme PASCAL, pour chacun des problèmes suivants :

- 1) permuter entre les deux variables X et Y ?
- 2) permuter entre les trois variables X, Y et Z de telle sorte que la valeur de X soit dans Y, celle de Y dans Z et la valeur de Z dans X ?
- 3) calculer la division entre deux nombres réels a et b ?
- 4) calculer la division euclidienne de deux nombres entiers a et b ?
- 5) calculer la somme de a et b et le produit de b et c ?
- 6) calculer la valeur absolue, le carré et la racine carrée d'un nombre ?

# Solution

## EXERCICE N°01 : ALGORITHMES → PROGRAMME

### 1- Traduire l'algorithme en Programme PASCAL

Algorithme	Programme PASCAL
<p><u>Algorithme</u> Exo1;</p> <p><u>Constantes</u> Pi=3.14;</p> <p><u>Variables</u> R,P,S : réel;</p> <p><u>Début</u></p> <p><i>{Entrées}</i> Écrire('Donner la valeur du rayon R :'); Lire(R);</p> <p><i>{Traitement}</i> P ← 2*PI*R; S ← PI*R*R;</p> <p><i>{Sorties}</i> Écrire('Périmètre = ', P); Écrire('Surface = ', S);</p> <p><u>Fin.</u></p>	<p><u>Program</u> Exo1;</p> <p><u>Const</u> Pi=3.14;</p> <p><u>Var</u> R,P,S : real;</p> <p><u>Begin</u></p> <p><i>{Entrées}</i> Write('Donner la valeur du rayon R :'); Read(R);</p> <p><i>{Traitement}</i> P := 2*PI*R; S := PI*R*R;</p> <p><i>{Sorties}</i> Writeln('Périmètre = ', P); Write('Surface = ', S);</p> <p><u>End.</u></p>

Cette version du programme est disponible, pour consultation et exécution, sur le lien suivant : <https://onlinegdb.com/DXMjtgZKt>

### 2- Formatage de l'affichage des valeurs réelles

Dans l'exécution du programme précédent, nous aurons l'affichage des variables P et S avec la virgule flottante, comme suit :

```
Donner la valeur du rayon R:5.36
Périmètre = 3.36608000000000002E+001
Surface = 9.02109440000000012E+001
```

Si on modifie les deux instructions d'écriture comme suit :

```
Writeln('Périmètre = ', P:0:3);
Write('Surface = ', S:10:2);
```

Nous auront un affichage comme suit :

```
Donner la valeur du rayon R:5.36
Périmètre =33.661
Surface = 90.21
```

Les périmètre avec 0 (zéro) espaces et 3 chiffres après la virgule, et la surface avec 10 espace et 2 chiffres après la virgule. On appel ça : l'affichage formaté des variables réels.

### 3- Dérouler le programme (ou l'algorithme) pour R = 3

Pour le déroulement, nous utilisons un tableau contenant deux colonnes : Instructions et Variables. On peut aussi ajouter une troisième colonne pour l'affichage.

Instructions	Variables			Affichage
	R	P	S	
Écrire('Donner la valeur du rayon R :');	/	/	/	Donner la valeur du rayon R : _
Lire(R);	3	/	/	
$P \leftarrow 2 * \text{PI} * R$ ; $P \leftarrow 2 * 3.14 * 3 = 18.84$	3	18.84	/	
$S \leftarrow \text{PI} * R * R$ ; $S \leftarrow 3.14 * 3 * 3$	3	18.84	28.26	
Écrire('Périmètre = ', P);	//	//	//	Périmètre = 18.84
Écrire('Surface = ', S);	//	//	//	Surface = 28.26

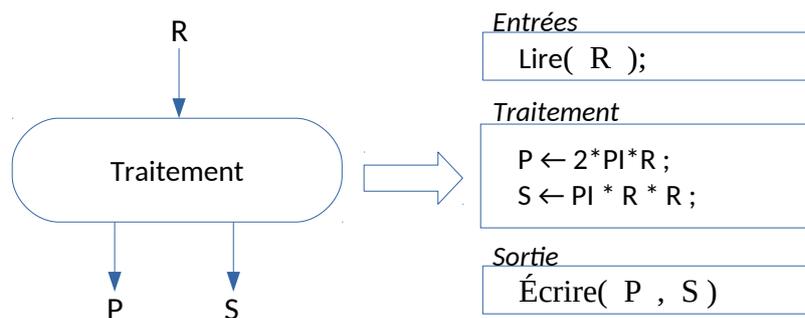
Un autre modèle de tableau de déroulement est le suivant :

Instructions	Variables		
	R	P	S
Écrire('Donner la valeur du rayon R :');	Donner la valeur du Rayon R : _		
Lire(R);	3	/	/
$P \leftarrow 2 * \text{PI} * R$ ; $P \leftarrow 2 * 3.14 * 3 = 18.84$	3	18.84	/
$S \leftarrow \text{PI} * R * R$ ; $S \leftarrow 3.14 * 3 * 3$	3	18.84	28.26
Écrire('Périmètre = ', P);	Périmètre = 18.84		
Écrire('Surface = ', S);	Surface = 28.26		

### 4- Compléter le schéma Entrées, Traitement et Sorties

Chaque algorithme possède des variables d'entrée, des variables de sorties, variables intermédiaire et une partie traitement :

- ✓ Les variables d'entrée sont les variables lues (l'instruction Lire)
- ✓ Les variables de sorties sont les variables affichées (l'instruction écrire)
- ✓ Les variables intermédiaire sont les variables non lues et non écrites
- ✓ Traitement : contient les instructions d'affectation, tests et les boucles.



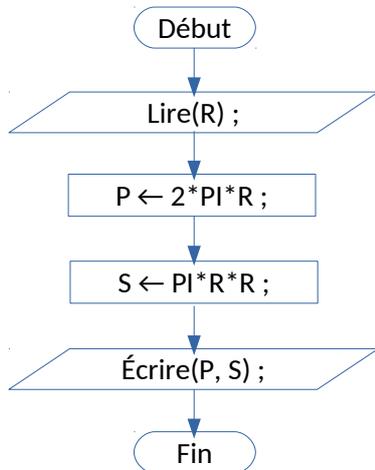
Dans cette série de TP, la partie de traitement contient uniquement l'instruction d'affectation :  $\leftarrow$  /  $:=$

## 5- Remarque (L'organigramme)

Un algorithme peut être schématisé sous forme d'un organigramme (voir le cours et la vidéo suivante :

<https://www.youtube.com/watch?v=oYfl4YAqnc0&list=PLwHHAvorm5F9Ay8o1tbtLaCWc3u9N-QRB&index=7> ).

Pour l'algorithme précédent, voici son organigramme :



Un organigramme est une façon d'illustrer l'acheminement d'exécution des différentes instructions. Ça illustre le déroulement d'un algorithme. Chaque instruction possède une forme géométrique. Le rectangle pour les affectation ou traitement, les losange pour les tests, les flèche pour l'acheminement d'exécution, ...

Dans l'organigramme ci à gauche, nous voyons clairement la séquentialité des instructions de lecture, écriture et affectation.

## EXERCICE N°02 : ÉNONCÉ DE PROBLÈME → ALGORITHME → PROGRAMME

Écrire un algorithme, puis traduit le en programme PASCAL, pour chacun des problèmes suivants :

### 1 – Permuter entre les deux variables X et Y ?

#### Analyse du problème :

On prends un exemple, pour illustrer (et clarifier) le problème :

soit  $x = 65$  et  $y = 17$ , à la fin de l'algorithme, nous devons avoir  $x = 17$  et  $y=65$

Variables d'entrée :  $x$  et  $y$  ( la première instruction : lire( $x,y$ ); )

Variables de sortie :  $x$  et  $y$  ( la dernière instruction : écrire( $x,y$ ); )

Traitement : échanger les valeurs de  $x$  et  $y$

La solution, triviale, qui vient à l'esprit, est :

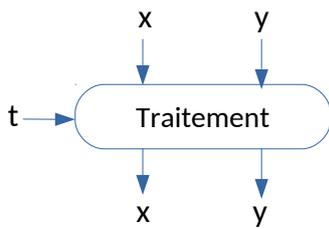
$x \leftarrow y$  ;

$y \leftarrow x$  ;

Le problème ici, est que la valeur initiale de  $x$  sera perdue, comme suit :

Instructions	Variables	
	x	y
Lire (x, y)	65	17
$x \leftarrow y$	17	17
$y \leftarrow x$	17	17
Écrire (x,y)	17	17

Le problème, avec cette solution, et que la valeur initiale de x est perdue (écrasée), comme montré dans le déroulement précédent.



L'idée est d'utiliser une troisième variable pour sauvegarder la valeur de x avant qu'elle soit écrasée. Pour échanger (permuter) les deux variable de x et y, on utilise une troisième variable t qui permet de conserver, par exemple la valeur de x, comme suit :

```
t ← x ;
x ← y ;
y ← t ; {t contient l'ancienne valeur de x}
```

Algorithmme
<b>Algorithmme</b> Exo2_1;
<b>Variables</b> x, y, t : entier;
<b>Début</b>
<i>{Entrées}</i> Lire(x, y) ;
<i>{Traitement}</i> t ← x ; x ← y ; y ← t ;
<i>{Sorties}</i> Écrire(x, y) ;
<b>Fin.</b>

Programme PASCAL
<b>Program</b> Exo2_1;
<b>Var</b> x,y,t : integer;
<b>Begin</b>
<i>{Entrées}</i> Read(x,y) ;
<i>{Traitement}</i> t := x ; x := y ; y := t ;
<i>{Sorties}</i> Write(x, y) ;
<b>End.</b>

Exemple de déroulement pour x = -20 et y=108 :

Instructions	Variables		
	x	y	t
Lire (x, y)	-20	108	/
t ← x	"	"	-20
x ← y	108	"	"
y ← t	108	-20	"
Écrire (x,y)	108	-20	"

## 2 – Permuter entre les trois variables X, Y et Z ?

### Analyse du problème :

Dans la permutation entre trois variables, X, Y et Z, nous aurons plusieurs cas, possibles. Pour cela, dans l'énoncé de la question N° 02, il y a une précision dans le sens de permutation :

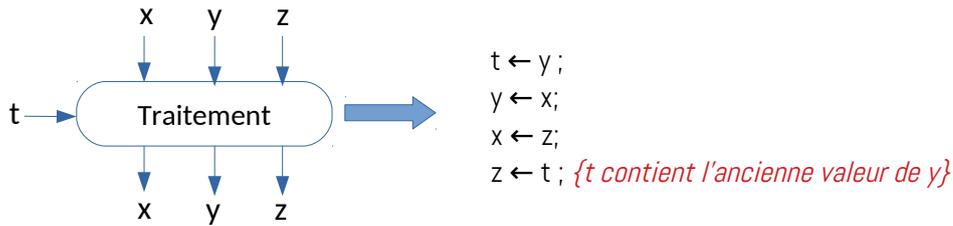
*la valeur de X soit dans Y, celle de Y dans Z et la valeur de Z dans X*

C'est-à-dire :

```
Y ← X ; X ← Z ; Z ← Y ;
```

De la même façon que le problème précédent, nous devons utiliser une 4ème variable T pour sauvegarder la valeur de Y (puisque Y est le premier à modifier dans la première affectation). Ainsi la solution devient :

$T \leftarrow Y ; Y \leftarrow X ; X \leftarrow Z ; Z \leftarrow T ;$  *{T contient l'ancienne valeur de Y avant qu'elle soit modifiée}*



Algorithme
<b>Algorithme</b> Exo2_2;
<b>Variables</b> x, y, z, t : entier;
<b>Début</b>
<i>{Entrées}</i> Lire(x, y, z);
<i>{Traitement}</i> t ← y; y ← x; x ← z; z ← t;
<i>{Sorties}</i> Écrire(x, y, z);
<b>Fin.</b>

Programme PASCAL
<b>Program</b> Exo2_2;
<b>Var</b> X, y, z, t : integer;
<b>Begin</b>
<i>{Entrées}</i> Read(x, y, z);
<i>{Traitement}</i> t := y; y := x; x := z; z := t;
<i>{Sorties}</i> Write(x, y, z);
<b>End.</b>

Dérouler l'algorithme (ou bien le programme) pour X=1, Y=20 et Z=33

Instructions	Variables			
	x	y	z	t
Lire (x, y, z)	1	20	33	/
t ← y	"	"	"	20
y ← x	"	1	"	"
x ← z	33	"	"	"
z ← t	"	"	20	"
Écrire (x,y, z)	33	1	20	"

**3 – Calculer la division entre deux variables réelles a et b**

**Analyse du problème :**



<u>Algorithme</u>
<b>Algorithme</b> Exo2_3;
<b>Variables</b> a, b, c : réel;
<b>Début</b>
<i>{Entrées}</i> Lire(a, b) ;
<i>{Traitement}</i> $c \leftarrow a / b$ ;
<i>{Sorties}</i> Écrire(b) ;
<b>Fin.</b>

<u>Programme PASCAL</u>
<b>Program</b> Exo2_3;
<b>Var</b> a, b, c : real;
<b>Begin</b>
<i>{Entrées}</i> Read(a, b) ;
<i>{Traitement}</i> t := a / b;
<i>{Sorties}</i> Write(b) ;
<b>End.</b>

Dérouler l'algorithme pour a=10 et b=2.5

<b>Instructions</b>	<b>Variables</b>		
	<b>a</b>	<b>b</b>	<b>c</b>
Lire (a, b)	10	2.5	/
$c \leftarrow a / b = 10 / 2.5 = 4$	"	"	4
Écrire (c)	"	"	<b>4</b>

**Remarque :**

L'algorithme précédent n'est pas complet, puisque la valeur de b n'a pas été vérifiée : la valeur de b doit être différente de 0 pour pouvoir réaliser la division a / b.

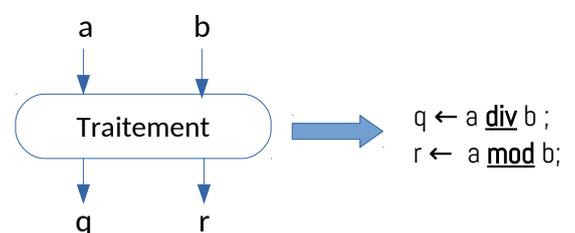
**4 – Calculer la division euclidienne entre deux entiers a et b**

**Analyse du problème :**

La division euclidienne entre deux entiers a et b est écrite écrite comme suit :

$a = b * q + r$ , tel-que q est quotient et r et le reste de division.

Donc, nous aurons le schéma d'entrée, sortie et traitement ci à droite:



<u>Algorithme</u>
<b>Algorithme</b> Exo2_4;
<b>Variables</b> a, b, q, r : entier;
<b>Début</b>
<i>{Entrées}</i> Lire(a, b) ;
<i>{Traitement}</i> $q \leftarrow a \text{ div } b$ ; $r \leftarrow a \text{ mod } b$ ;
<i>{Sorties}</i> Écrire(q, r) ;
<b>Fin.</b>

<u>Programme Pascal</u>
<b>Program</b> Exo2_4;
<b>Var</b> a, b, q, r : integer;
<b>Début</b>
<i>{Entrées}</i> Read(a, b) ;
<i>{Traitement}</i> $q := a \text{ div } b$ ; $r := a \text{ mod } b$ ;
<i>{Sorties}</i> Write(q, r) ;
<b>Fin.</b>

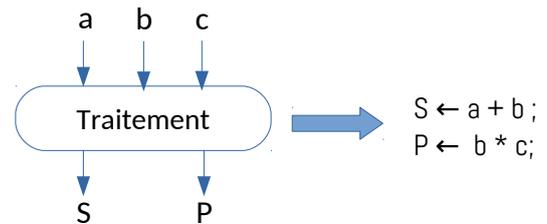
## 5 – Calculer la somme de a et b et le produit de b et c

### Analyse du problème :

La somme de a et b sera enregistré dans la variable S, donc on écrit :  $S \leftarrow a + b$ ;

Le produit de b et c sera stocké dans la variable P, on écrit alors :  $P \leftarrow b * c$  ;

Donc, nous aurons le schéma d'entrée, sortie et traitement ci à droite:



<u>Algorithme</u>
<b>Algorithme</b> Exo2_5;
<b>Variables</b> a, b, c S, P : entier;
<b>Début</b>
<i>{Entrées}</i> Lire(a, b, c) ;
<i>{Traitement}</i> S ← a + b ; P ← b * c ;
<i>{Sorties}</i> Écrire(S, P) ;
<b>Fin.</b>

<u>Programme Pascal</u>
<b>Program</b> Exo2_5;
<b>Var</b> a, b, c, S, P : integer;
<b>Début</b>
<i>{Entrées}</i> Read(a, b, c) ;
<i>{Traitement}</i> S := a + b ; P := a * b ;
<i>{Sorties}</i> Write(S, P) ;
<b>Fin.</b>

## 6– Calculer la valeur absolue, le carré et la racine carrée d'un nombre

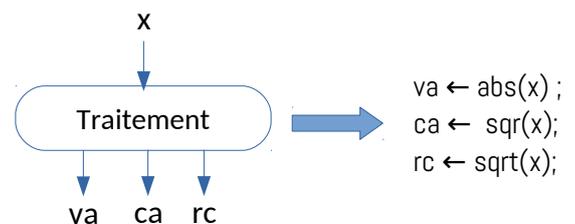
### Analyse du problème :

Soit x un nombre réel, sa valeur absolue va sera obtenue par l'affectation :  $va \leftarrow \text{abs}(x)$  ;

Son carré, ca, sera calculé par :  $ca \leftarrow \text{sqr}(x)$  ;

Sa racine carrée, rc, est obtenue par l'affectation :  $rc \leftarrow \text{sqrt}(x)$ ;

Ainsi, nous obtenons le schéma d'entrée, sortie et traitement ci à droite:



<u>Algorithme</u>
<b>Algorithme</b> Exo2_6;
<b>Variables</b> x, va, ca, rc : réel;
<b>Début</b>
<i>{Entrées}</i> Lire(x) ;
<i>{Traitement}</i> va ← abs(x); ca ← sqr(x); rc ← sqrt(x);
<i>{Sorties}</i> Écrire(va, ca, rc) ;
<b>Fin.</b>

<u>Programme Pascal</u>
<b>Program</b> Exo2_6;
<b>Var</b> x, va, ca, rc : real;
<b>Début</b>
<i>{Entrées}</i> Read(x) ;
<i>{Traitement}</i> va := abs(x); ca := sqr(x); rc := sqrt(x);
<i>{Sorties}</i> Write(va, ca, rc) ;
<b>Fin.</b>

---

# ***Bon Courage & Travaillez bien.***

---

Cours Elearning :

<https://elearning.univ-bejaia.dz/course/view.php?id=7944>

Page facebook :

<https://www.facebook.com/InitiationAlgoProgrammation/>

La chaîne Youtube :

<https://www.youtube.com/c/AlgoProgrammation1èreAnnéeTechnologie>

---

Adapté par: Redouane OUZEGGANE  
[rouzeggane@gmail.com](mailto:rouzeggane@gmail.com) - [redouane.ouzeggane@univ-bejaia.dz](mailto:redouane.ouzeggane@univ-bejaia.dz)