

## TP Structure des ordinateurs et applications

### Corrigé de la série de TP N°3

#### Rappel :

##### *Structure d'un algorithme / programme*

Un algorithme manipule des données, les données avant de les utiliser il faut les identifier et les déclarer en utilisant les identificateurs. Un algorithme est constitué de trois parties :

- **Entête** : dans cette partie on déclare le nom de l'algorithme à travers un identificateur.
- **Déclarations** : dans cette partie on déclare toutes les données utilisées par l'algorithme.
- **Corps** : représente la séquence d'actions (instructions)

Pour écrire un algorithme, il faut suivre la structure suivante :

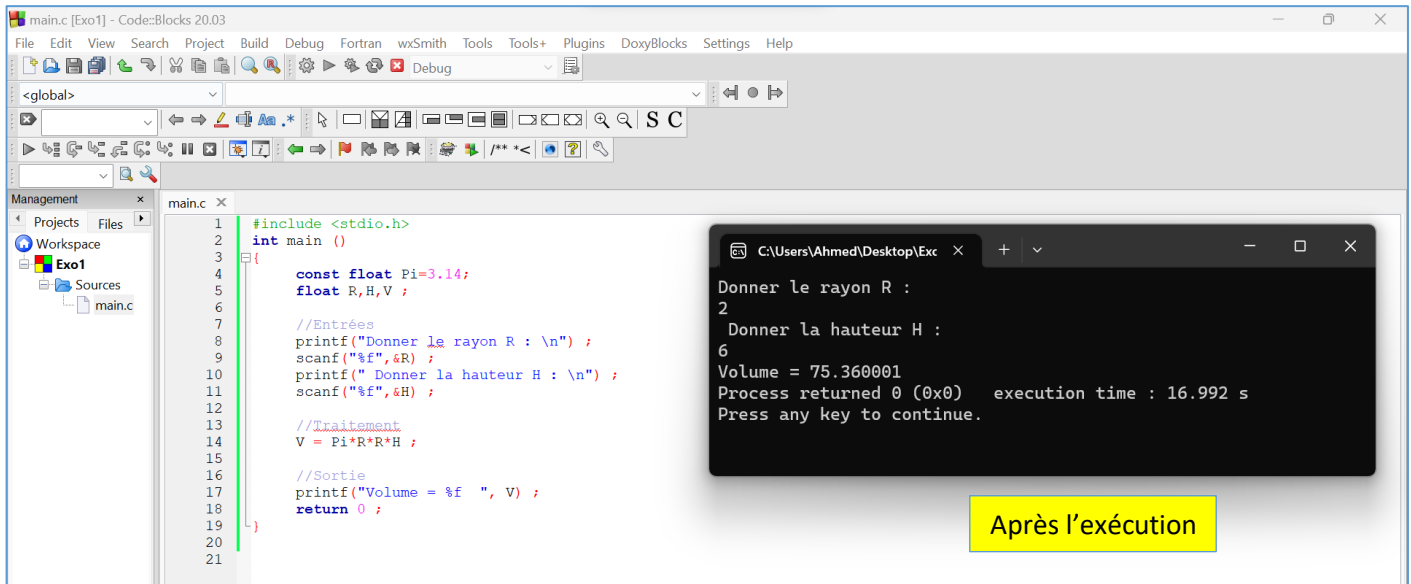
<p><b>Algorithme</b> &lt;id_algorithme&gt;                  &lt;Déclarations&gt;  <b>Début</b>                  &lt;Corps : Instructions&gt;  <b>Fin.</b></p>	<pre>#include &lt;stdio.h&gt; int main() {     &lt;déclarations&gt;;     &lt;Instructions&gt;;     return 0; }</pre>
---	--

#### Exercice N°01 : (Algorithme → Programme C)

1) Traduire l'algorithme en Programme C

Algorithme	Programme C
<p><b>Algorithme</b> Exo1;</p> <p><b>Constantes</b>                  Pi=3.14 ;</p> <p><b>Variables</b>                  R,H,V : Réel;</p> <p><b>Début</b></p> <p style="color: red;">//Entrées</p> <p>Écrire("Donner le rayon R : ");                  Lire(R);                  Écrire("Donner la hauteur H :");                  Lire(H);</p> <p style="color: red;">//Traitement</p> <p>V ← Pi*R*R*H ;</p> <p style="color: red;">//Sortie</p> <p>Écrire("Volume = ", V);</p> <p><b>Fin.</b></p>	<pre>#include &lt;stdio.h&gt; int main () {     const float Pi=3.14;     float R,H,V ;      //Entrées     printf("Donner le rayon R : \n") ;     scanf("%f",&amp;R) ;     printf(" Donner la hauteur H : \n") ;     scanf("%f",&amp;H) ;      //Traitement     V = Pi*R*R*H ;      //Sortie     printf("Volume = %f ", V) ;     return 0 ; }</pre>

## 2) Compiler et exécuter le programme pour : R = 2 et H=6



```
1 #include <stdio.h>
2 int main ()
3 {
4     const float Pi=3.14;
5     float R,H,V ;
6
7     //Entrées
8     printf("Donner le rayon R : \n") ;
9     scanf("%f",&R) ;
10    printf(" Donner la hauteur H : \n") ;
11    scanf("%f",&H) ;
12
13    //Traitement
14    V = Pi*R*R*H ;
15
16    //Sortie
17    printf("Volume = %f ", V) ;
18    return 0 ;
19 }
20
21
```

Donner le rayon R :  
2  
Donner la hauteur H :  
6  
Volume = 75.360001  
Process returned 0 (0x0) execution time : 16.992 s  
Press any key to continue.

Après l'exécution

## 3) Remplacer la dernière instruction :

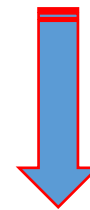
Écrire("Volume = ", V), par une autre instruction qui permet d'afficher le volume avec seulement deux chiffres après la virgule,

```
Programme C
#include <stdio.h>
int main ()
{
    const float Pi=3.14;
    float R,H,V ;

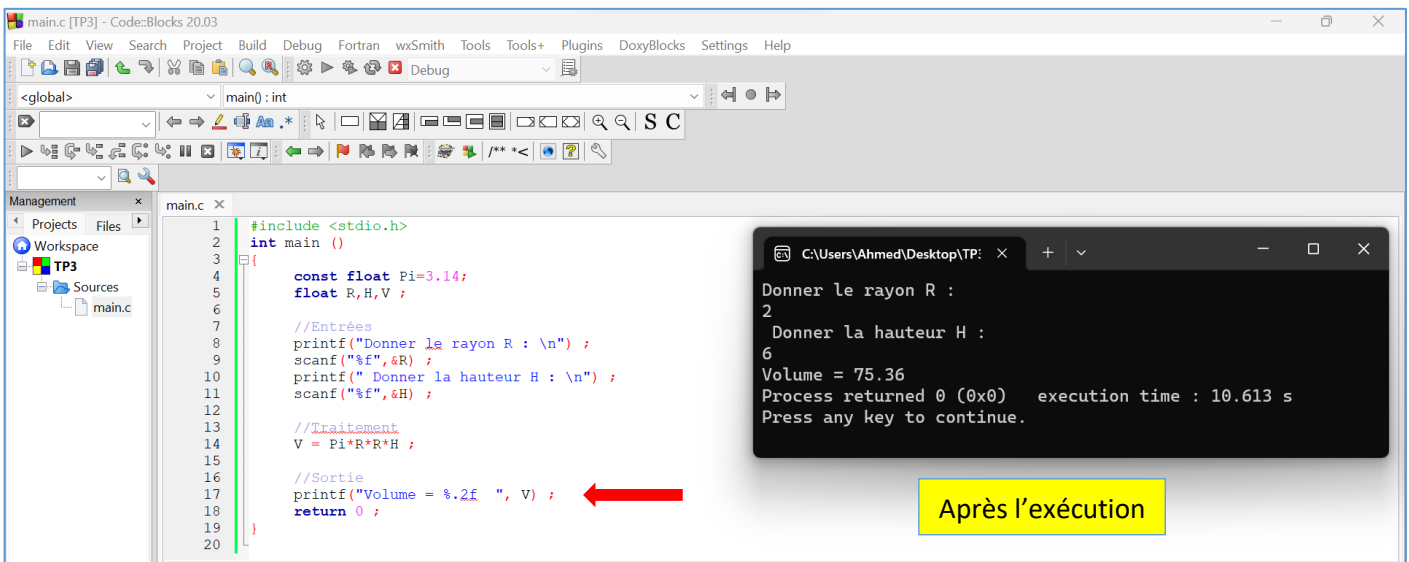
    //Entrées
    printf("Donner le rayon R : \n") ;
    scanf("%f",&R) ;
    printf(" Donner la hauteur H : \n") ;
    scanf("%f",&H) ;

    //Traitement
    V = Pi*R*R*H ;

    //Sortie
    printf("Volume = %.2f ", V) ;
    return 0 ;
}
```



**printf("Volume = %.2f ", V) ;**  
Où 2 représente deux chiffres après la virgule.



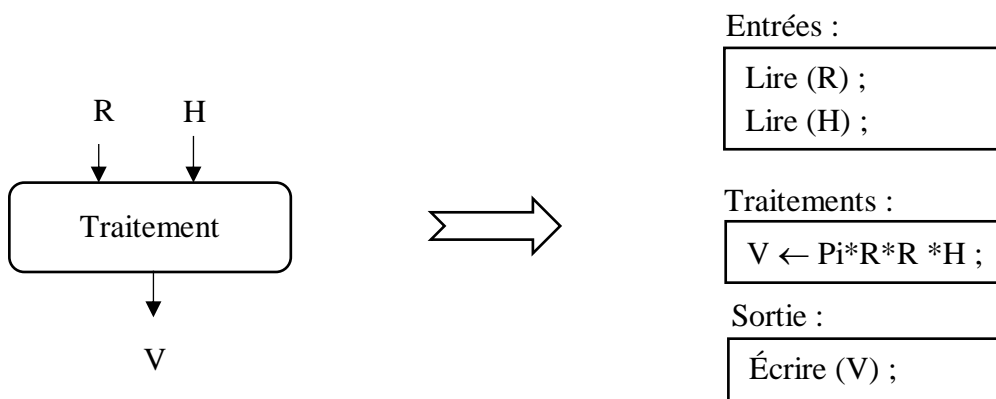
4) Déroulement de l'algorithme pour R=2 et H=6

Instructions	Variables			Affichage
	R	H	V	
Écrire("Donner le rayon R :")	/	/	/	Donner le rayon R :
Lire (R)	2	/	/	
Écrire("Donner la hauteur H :")	2	/	/	Donner la hauteur H :
Lire (H)	2	6	/	
$V \leftarrow \text{Pi} * R * R * H$	2	6	75.360001	
Ecrire ("Volume =", P) ;	2	6	75.360001	Volume = 75.360001

5) Dédire ce que fait le programme ?

Le programme calcule le volume d'un cylindre

6) Compléter le schéma suivant :



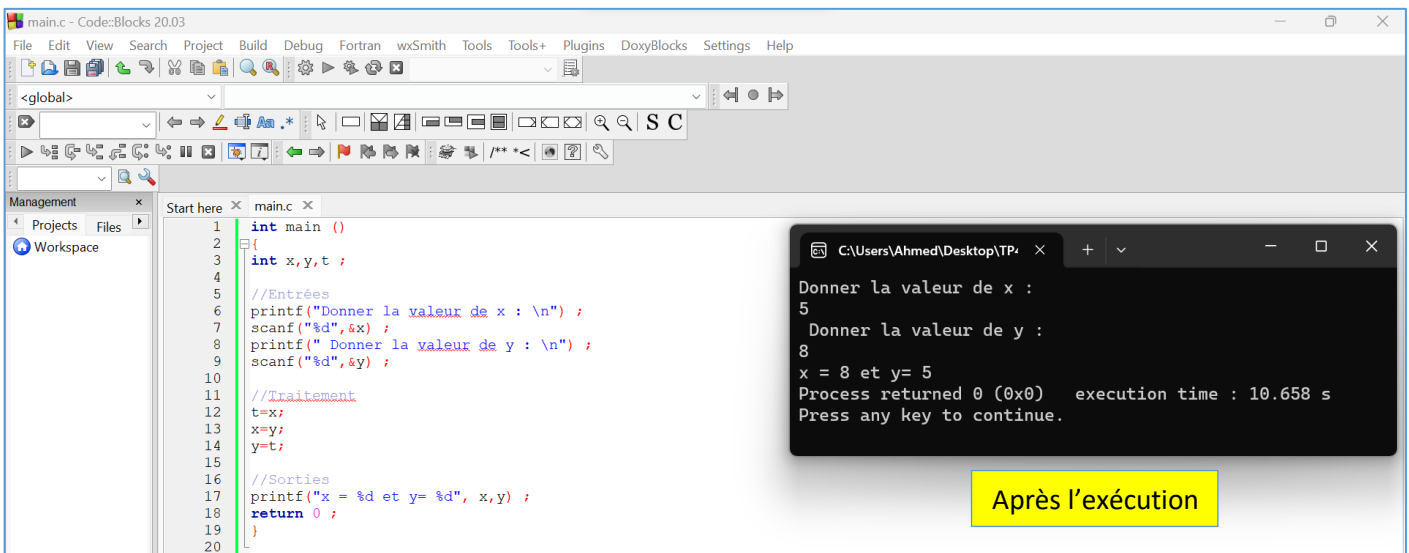
Chaque algorithme possède des variables d'entrée, des variables de sorties, constantes et une partie du traitement :

- Les variables d'entrée sont les variables lues (l'instruction **Lire**) ;
- Les variables de sorties sont les variables affichées (l'instruction **Ecrire**) ;
- Les données intermédiaires qui peuvent être des variables ou des constantes (dans notre cas **Pi**) qui sont des données non lues et non écrites utilisées pendant le traitement ;
- Traitement : contient les instructions d'affectation, tests et les boucles.

**Exercice N°02 :** (Enoncé du problème → Algorithme → Programme C)

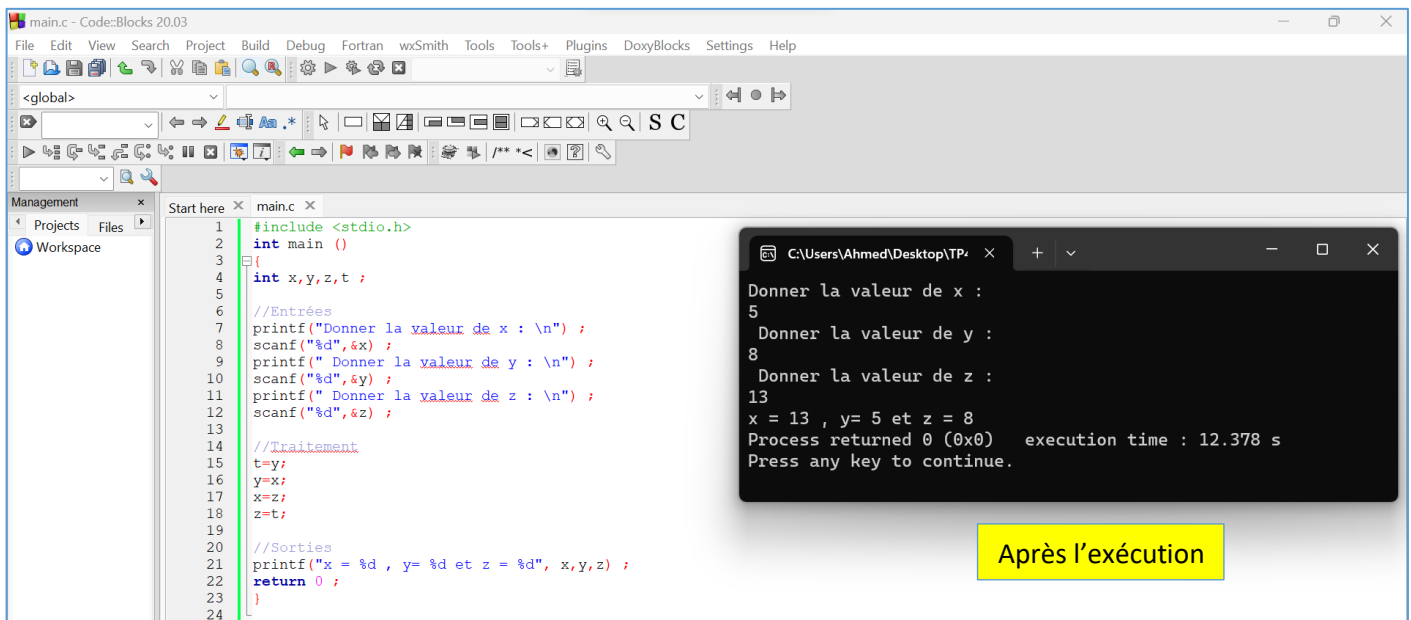
1) Permuter entre les deux variables X et Y ?

Algorithme	Programme C
Algorithme Exo2_1; <b>Variabes</b> x, y, t : <b>entier</b> ;  <b>Début</b> //Entrées <b>Ecrire</b> ("Donner la valeur de x : ") ; <b>Lire</b> (x) ; <b>Ecrire</b> ("Donner la valeur de y : ") ; <b>Lire</b> (y) ;  //Traitement t ← x ; x ← y ; y ← t ;  //Sorties Écrire("x=", x, "y=", y) ; <b>Fin.</b>	<pre> <b>int</b> main () {     <b>int</b> x,y,t ;      //Entrées     <b>printf</b>("Donner la valeur de x : \n") ;     <b>scanf</b>("%d",&amp;x) ;     <b>printf</b>(" Donner la valeur de y : \n") ;     <b>scanf</b>("%d",&amp;y) ;      //Traitement     t=x;     x=y;     y=t;      //Sorties     <b>printf</b>("x = %d et y= %d", x,y) ;     <b>return</b> 0 ; }           </pre>



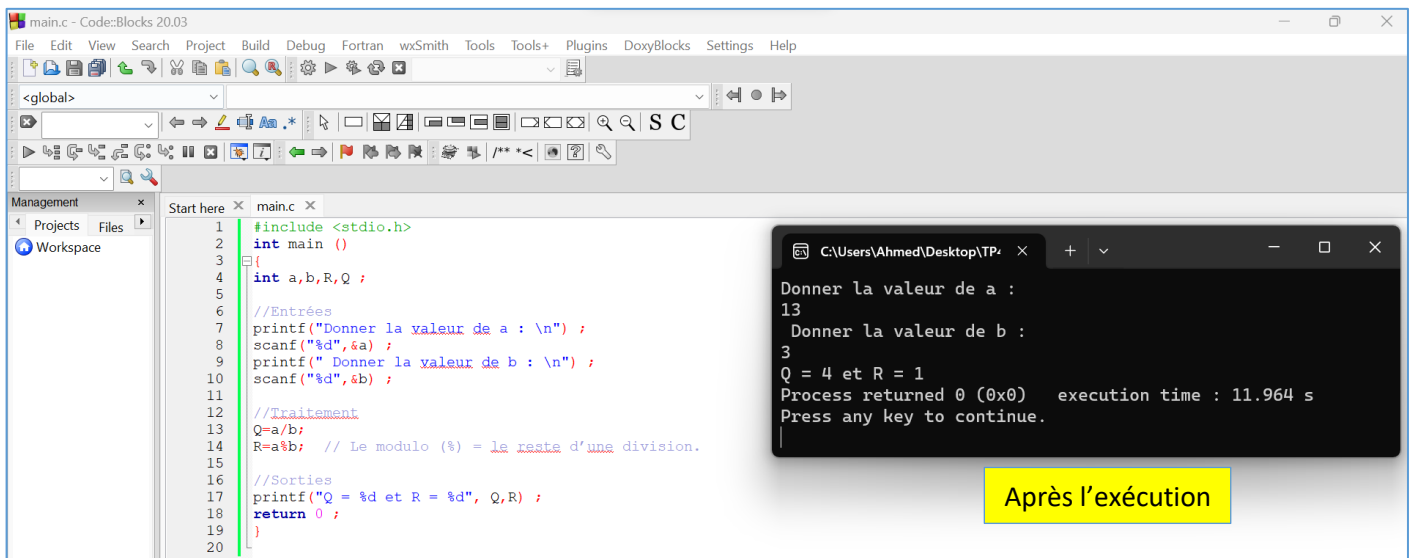
2) Permuter entre les trois variables X, Y et Z de telle sorte que la valeur de X soit dans Y, celle de Y dans Z et la valeur de Z dans X ?

Algorithme	Programme C
<p><b>Algorithme Exo2_2;</b></p> <p><b>Variables</b> x, y, z, t : entier;</p> <p><b>Début</b></p> <p><i>//Entrées</i> Ecrire("Donner la valeur de x : "); Lire(x); Ecrire("Donner la valeur de y : "); Lire(y); Ecrire("Donner la valeur de z : "); Lire(z);</p> <p><i>//Traitement</i> t ← y; y ← x; x ← z; z ← t;</p> <p><i>//Sorties</i> Écrire("x=", x, "y=", y, "z=",z);</p> <p><b>Fin.</b></p>	<pre>#include &lt;stdio.h&gt; int main () {     int x,y,z,t ;      //Entrées     printf("Donner la valeur de x : \n") ;     scanf("%d",&amp;x) ;     printf(" Donner la valeur de y : \n") ;     scanf("%d",&amp;y) ;     printf(" Donner la valeur de z : \n") ;     scanf("%d",&amp;z) ;      //Traitement     t=y;     y=x;     x=z;     z=t;      //Sorties     printf("x = %d , y= %d et z = %d", x,y,z) ;     return 0 ; }</pre>



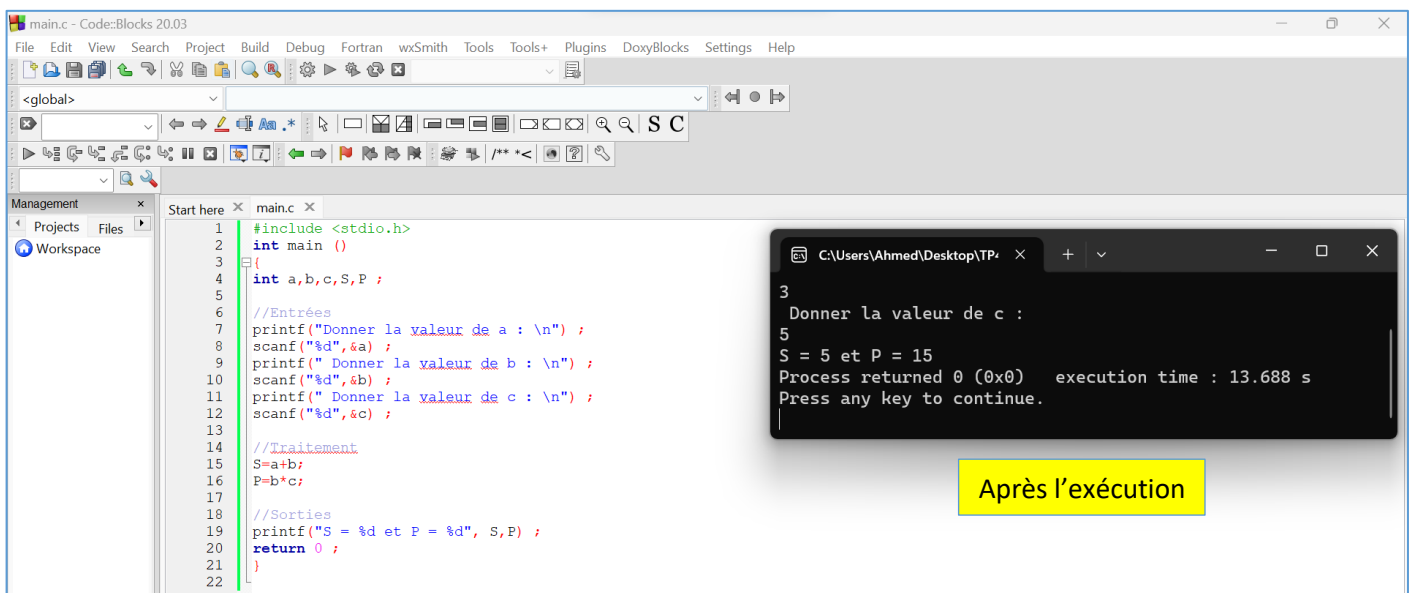
3) Calculer le quotient et le reste de la division euclidienne de a par b ?

Algorithme	Programme C
<p><b>Algorithme Exo2_4;</b></p> <p><b>Variabes</b> a, b, Q, R : entier;</p> <p><b>Début</b></p> <p><b>//Entrées</b> Ecrire("Donner la valeur de a : "); Lire(a); Ecrire("Donner la valeur de b : "); Lire(b);</p> <p><b>//Traitement</b> Q ← a div b; //div : division entière R ← a mod b; //mod : reste de division</p> <p><b>//Sorties</b> Écrire("Q=", Q, " et R= ", R);</p> <p><b>Fin.</b></p>	<pre> #include &lt;stdio.h&gt; int main () {     int a,b,R,Q ;      //Entrées     printf("Donner la valeur de a : \n");     scanf("%d",&amp;a) ;     printf(" Donner la valeur de b : \n");     scanf("%d",&amp;b) ;      //Traitement     Q=a/b;     R=a%b; // Le modulo (%) = le reste d'une division.      //Sorties     printf("Q = %d et R = %d", Q,R) ;     return 0 ; } </pre>



4) Calculer la somme de a et b et le produit de b et c ?

Algorithme	Programme C
<p><b>Algorithme Exo2_5;</b></p> <p><b>Variables</b> a, b, c, S, P : <b>entier;</b></p> <p><b>Début</b></p> <p><b>//Entrées</b>  <b>Ecrire</b>("Donner la valeur de a : ") ;  <b>Lire</b>(a) ;  <b>Ecrire</b>("Donner la valeur de b : ") ;  <b>Lire</b>(b) ;  <b>Ecrire</b>("Donner la valeur de c : ") ;  <b>Lire</b>(c) ;</p> <p><b>//Traitement</b>  S ← a + b;  P ← b * c;</p> <p><b>//Sorties</b>  <b>Écrire</b>("S=", S, "P= ", P) ;</p> <p><b>Fin.</b></p>	<pre> #include &lt;stdio.h&gt; int main () {     int a,b,c,S,P ;      //Entrées     printf("Donner la valeur de a : \n") ;     scanf ("%d",&amp;a) ;     printf(" Donner la valeur de b : \n") ;     scanf ("%d",&amp;b) ;     printf(" Donner la valeur de c : \n") ;     scanf ("%d",&amp;c) ;      //Traitement     S=a+b;     P=b*c;      //Sorties     printf("S = %d et P = %d", S,P) ;     return 0 ; } </pre>

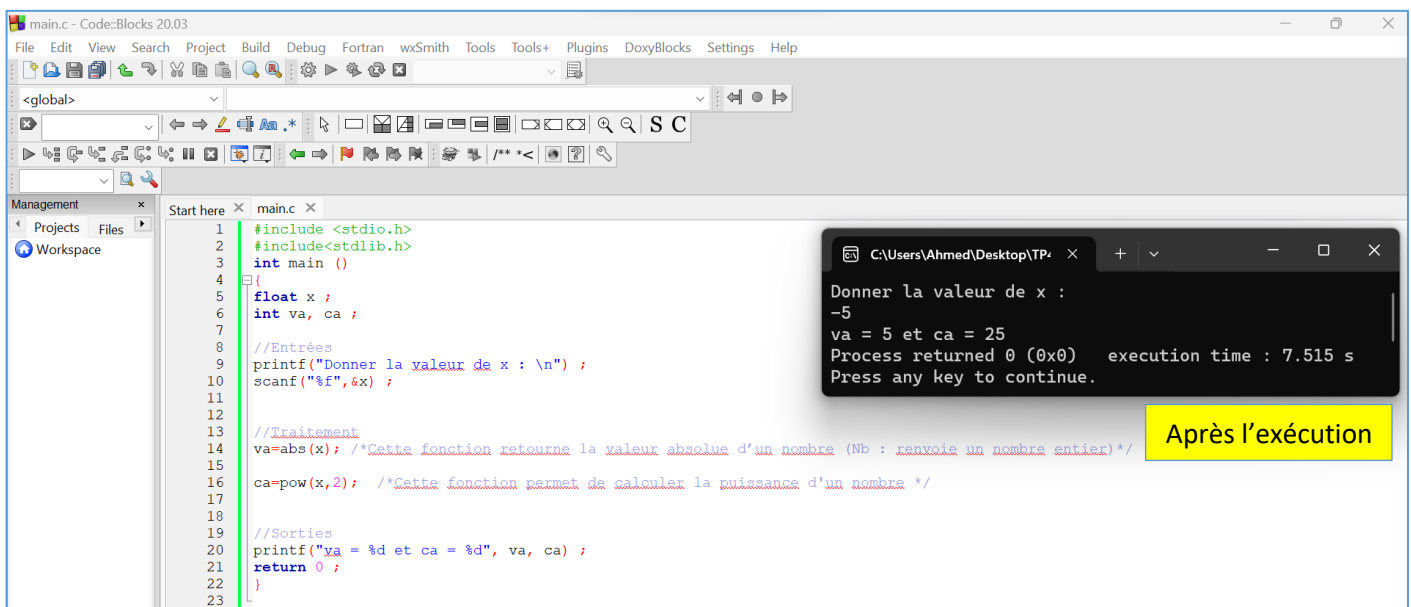


Après l'exécution

5) Calculer la valeur absolue et le carré d'un nombre ?

Algorithme	Programme C
<p><b>Algorithme</b> Exo2_6;</p> <p><b>Variables</b></p> <p>x : réel;</p> <p>va, ca : entier;</p> <p><b>Début</b></p> <p><i>//Entrées</i></p> <p><b>Ecrire</b>("Donner la valeur de x : ");</p> <p><b>Lire</b>(x);</p> <p><i>//Traitement</i></p> <p>va ←  x ;</p> <p>ca ← x<sup>2</sup>;</p> <p><i>//Sorties</i></p> <p><b>Écrire</b>("va=", va, "ca=", ca);</p> <p><b>Fin.</b></p>	<pre>#include &lt;stdio.h&gt; #include&lt;stdlib.h&gt; int main () {     float x ;     int va, ca ;      <i>//Entrées</i>     printf("Donner la valeur de x : \n");     scanf("%f",&amp;x) ;      <i>//Traitement</i>     va=abs(x); <i>/*Cette fonction retourne la valeur absolue d'un nombre (Nb : renvoie un nombre entier)*/</i>      ca=pow(x,2); <i>/*Cette fonction permet de calculer la puissance d'un nombre */</i>      <i>//Sorties</i>     printf("va = %d et ca = %d", va, ca) ;     return 0 ; }</pre>





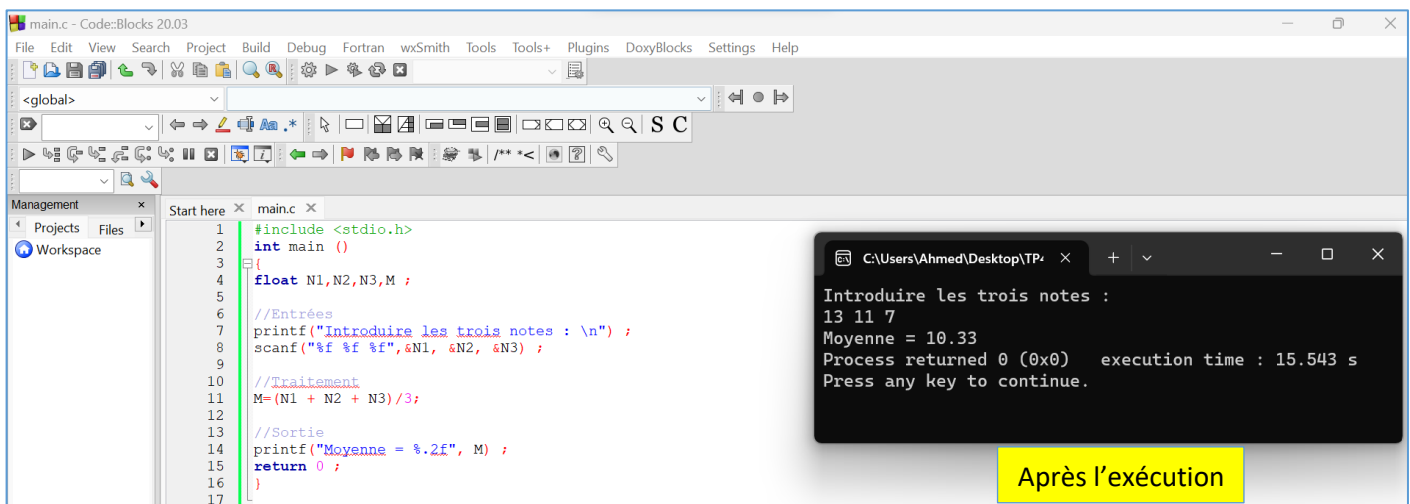
## 6) Convertir en octets un nombre donné en bits ?

Algorithme	Programme C
<p><b>Algorithme Exo2_7;</b></p> <p><b>Variables</b> bit, octet : réel;</p> <p><b>Début</b></p> <p><i>//Entrée</i> Écrire("Nombres de bits ="); Lire(bit);</p> <p><i>//Traitement</i> octet ← bit/8;</p> <p><i>//Sortie</i> Écrire(bit, " bits =", octet, " octet");</p> <p><b>Fin.</b></p>	<pre> #include &lt;stdio.h&gt; int main () {     float bit, octet;      //Entrée     printf("Nombres de bits : \n");     scanf("%f", &amp;bit);      //Traitement     octet=bit/8;      //Sortie     printf("%f bits = %f octet", bit, octet);     return 0 ; } </pre>



7) Lire les notes de trois matières (N1, N2 et N3) ensuite calculer et afficher leur moyenne M

Algorithme	Programme C
<p><b>Algorithme</b> Exo2_7_a;</p> <p><b>Variables</b> N1, N2, N3, M : réel ;</p> <p><b>Début</b> //Entrées Écrire ("Introduire les trois notes : "); Lire(N1, N2, N3);</p> <p>//Traitement <math>M \leftarrow (N1 + N2 + N3)/3</math>;</p> <p>//Sorties Écrire("Moyenne = ", M:0:2);</p> <p><b>Fin.</b></p>	<pre>#include &lt;stdio.h&gt; int main () {     float N1,N2,N3,M ;      //Entrées     printf("Introduire les trois notes : \n");     scanf("%f %f %f",&amp;N1, &amp;N2, &amp;N3);      //Traitement     M=(N1 + N2 + N3)/3;      //Sortie     printf("Moyenne = %.2f", M);     return 0 ; }</pre>



Modifier l'algorithme dans le cas où des coefficients (C1, C2 et C3) sont attribués aux trois matières.

Algorithme	Programme C
<p><b>Algorithme</b> Exo2_7_b;</p> <p><b>Variables</b> N1, N2, N3, M : réel ; C1, C2, C3 : entier ;</p> <p><b>Début</b> //Entrées Écrire ("Introduire les trois notes : "); Lire(N1, N2, N3);</p> <p>Écrire ("Introduire les trois coefficients : "); Lire(C1, C2, C3);</p>	<pre>#include &lt;stdio.h&gt; int main () {     float N1,N2,N3,M ;     int C1,C2,C3;      //Entrées     printf("Introduire les trois notes : \n");     scanf("%f %f %f",&amp;N1, &amp;N2, &amp;N3);      printf("Introduire les trois coefficients : \n");     scanf("%d %d %d",&amp;C1, &amp;C2, &amp;C3); }</pre>

**//Traitement**

$M \leftarrow (N1 * C1 + N2 * C2 + N3 * C3) / (C1 + C2 + C3);$

**//Sorties**

**Écrire("Moyenne =", M:0:2);**  
**Fin.**

**//Traitement**

$M = (N1 * C1 + N2 * C2 + N3 * C3) / (C1 + C2 + C3);$

**//Sortie**

**printf("Moyenne = %.2f", M);**  
**return 0;**  
**}**

The screenshot shows the Code::Blocks IDE with a C program in the editor and a terminal window showing the program's execution. The code in the editor is as follows:

```
1 #include <stdio.h>
2 int main ()
3 {
4     float N1,N2,N3,M ;
5     int C1,C2,C3;
6
7     //Entrées
8     printf("Introduire les trois notes : \n") ;
9     scanf("%f %f %f",&N1, &N2, &N3) ;
10
11    printf("Introduire les trois coefficients : \n") ;
12    scanf("%d %d %d",&C1, &C2, &C3) ;
13
14    //Traitement
15    M=(N1*C1 + N2*C2 + N3*C3) / (C1+C2+C3);
16
17    //Sortie
18    printf("Moyenne = %.2f", M) ;
19    return 0 ;
20 }
21
```

The terminal window shows the following output:

```
C:\Users\Ahmed\Desktop\TP: x + v - □ x
Introduire les trois notes :
12 14 6
Introduire les trois coefficients :
3 4 2
Moyenne = 11.56
Process returned 0 (0x0) execution time : 18.144 s
Press any key to continue.
```

A yellow box with the text "Après l'exécution" is positioned below the terminal window.