

4 Conditions

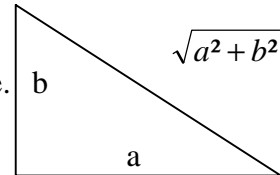
Avec des Si on mettrait Paris en bouteille...

4.1 Exercice de mise en bouche

Ecrire un programme qui met en application le théorème de Pythagore pour calculer l'hypoténuse d'un triangle rectangle.

Rappel :

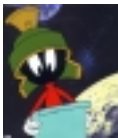
Dans un triangle rectangle, les longueurs côtés sont définies comme le montre la figure ci-contre.



Note :

- ≡ La racine carrée s'obtient par l'utilisation de la fonction **sqrt** (valeur) contenue dans la bibliothèque math.h. (`#include <math.h>`)
- ≡ a^2 peut s'obtenir par `a*a`.

Méthodologie :



1. Rechercher les variables nécessaires et les déclarer dans le programme.
2. Faire saisir a au clavier.
3. Faire saisir b au clavier.
4. Effectuer l'opération de la racine carrée et afficher le résultat.

4.2 Les conditions : Si Alors Sinon

if (condition vraie)	si (condition vraie)
{	{
instructions 1	alors
}	faire instructions 1
else	}
{	sinon
instructions 2	{
}	faire instructions 2
	}

Les conditions s'expriment avec des opérateurs logiques ...

4.2.1 Opérateurs logiques relationnels

Ils servent à comparer deux nombres entre eux.

<i>Libellé</i>	<i>Opérateur</i>
<i>Inférieur</i>	<
<i>Supérieur</i>	>
<i>Equivalent</i>	==
<i>Différent</i>	!=
<i>Inférieur ou égal</i>	<=
<i>Supérieur ou égal</i>	>=

4.2.2 Opérateurs logiques purs

Ce sont des opérateurs logiques permettant de combiner des expressions logiques.

<i>Libellé</i>	<i>Opérateur</i>
<i>Et (and)</i>	&&
<i>Ou (or)</i>	
<i>Non (not)</i>	!

N.B. | se nomme en informatique un pipe.

4.2.3 Vrai ou faux

La valeur Vrai peut être assimilée à la valeur numérique 1 ou à toute valeur > 0.

La valeur Faux peut être assimilée à la valeur numérique 0.

L'opérateur Ou (||) correspond alors à une addition

Ou	Vrai	Faux	+	1	0
Vrai	Vrai	Vrai	1	2	1
Faux	Vrai	Faux	0	1	0

L'opérateur Et (&&) correspond alors à une multiplication

Et	Vrai	Faux	*	1	0
Vrai	Vrai	Faux	1	1	0
Faux	Faux	Faux	0	0	0

On notera que !Vrai = Faux et !Faux = Vrai.

4.2.4 Combinaison

Toutes les opérations logiques peuvent se combiner entre elles. La seule condition d'utilisation d'un si (if) avec de telles combinaisons est de l'entourer de ().

Exemple :

```
if ((car == 'a') || (car == 'A'))
```

4.2.5 Astuce

Vous verrez souvent ce type de code écrit :

```
if (er)
{
    /* Alors faire quelque chose */
}
```

En appliquant ce qui a été vu précédemment, on en déduit que ce code signifie que

```
si (er != 0) /* si er différent de 0 */
{
    /* Alors faire quelque chose */
}
```

4.3 Les accolades

Les accolades entourant les blocs d'instructions d'une condition peuvent être omises si le bloc est constitué d'une seule instruction.

Exemple :

```
if (car == 'b')
    printf ("car vaut b.");
else
    printf ("car est différent de b.");
```

4.4 Exercices

! Faites saisir une variable de type entière et indiquez à l'utilisateur si celle-ci est positive ou négative ou nulle.

Aide :

```
if (a>0)
{
    printf ("Valeur positive");
}
else
{
    printf ("Valeur négative");
}
```

! Faites saisir une variable de type entière et indiquez à l'utilisateur si celle-ci est positive, négative ou nulle.

! Faites saisir une variable de type caractère et indiquez à l'utilisateur si celle-ci est une voyelle.

Corrections des exercices du chapitre 4

! *Ecrire un programme qui effectue le théorème de Pythagore*

```
#include <stdio.h>
#include <conio.h>
#include <math.h>

int main ()
{
    float a;    /* base du triangle */
    float b;    /* côté du triangle rectangle */
    float p;    /* valeur de l'hypoténuse (p pour Pythagore !) */

    /* Initialisation des variables pour palier aux erreurs */
    a = 0;
    b = 0;

    /* Effacer l'écran */
    clrscr ();

    /* Saisie de a */
    printf ("Valeur de la base : ");
    scanf ("%f",&a);

    /* Saisie de b */
    printf ("Valeur du côté : ");
    scanf ("%f",&b);

    /* Calcul par Pythagore */
    p = sqrt (a*a + b*b);

    /* Affichage du résultat */
    printf ("L'hypoténuse mesure : %.2f",p);

    /* Attendre avant de sortir */
    getch ();

    return (0);
}
```

! *Test du signe d'une valeur saisie au clavier*

```
#include <stdio.h>
#include <conio.h>
#include <math.h>

int main ()
{
    /* Valeur que l'on va saisir */
    int a = 0;

    /* Efface l'écran */
    clrscr ();

    /* Saisie de a */
    printf("Saisie de a : ");
```

```
scanf("%f",&a);

/* Test condition a<0 */
if (a<0)
{
    printf("la variable a est négative.\n");
}
else
{
    /* Test condition a>0 */
    if (a>0)
    {
        printf("la variable a est positive\n");
    }
    /* Sinon a est nulle */
    else
    {
        printf("la variable a est nulle\n");
    }
}

getch ();
return (0);
}
```

! **Test du signe d'une valeur saisie au clavier**

```
#include <stdio.h>
#include <conio.h>
#include <math.h>

int main ()
{
    /* Valeur que l'on va saisir */
    char car;

    /* Efface l'écran */
    clrscr ();

    /* Saisie du caractère a */
    printf("Saisie du caractère : ");
    scanf("%c",&car);

    /* Test condition car voyelle minuscule */
    if ((car == 'a') || (car == 'e') || (car == 'i') || (car == 'o') ||
        (car == 'u') || (car == 'y'))
    {
        printf("la variable car est une voyelle.\n");
    }
    else
    {
        printf("la variable car est une consonne.\n");
    }

    getch ();
    return (0);
}
```