

## 9 Les boucles

### 9.1 Et les shadoks pédalèrent pendant 15 tours ...

Pour faire effectuer un certain nombre de fois une tâche on utilise l'instruction `for` de la façon suivante (avec `i`, une variable de type entier (`int`)).

```
for (i=point de départ; i<point d'arrivée; i=i+pas)  
{  
    instruction(s) répétée(s);  
}
```

Pour un souci de simplicité, nous dirons simplement que la formule suivante :

```
for (i=0; i<15; i++)  
{  
    instr;  
}
```

signifie que l'on va exécuter *instr* pour `i` allant de 0 à 14 (<15) c'est à dire 15 fois.

*Exemple:*

```
#include <conio.h>  
  
int main ()  
{  
    int i;  
  
    clrscr ();  
  
    for (i=0; i<15; i++)  
    {  
        printf ("Je me répète pour i valant %d\n",i);  
    }  
  
    printf ("Je me suis répété, 15 fois");  
    getch ();  
}
```

### 9.2 Syntaxe

De la même façon que le **if** le **for** ne nécessite pas d'accolade si le nombre d'instructions à répéter est de 1.

*Exemple :*

On peut utiliser cette fonctionnalité dans le programme précédent en remplaçant :

```
for (i=0; i<15; i++)  
{  
    printf ("Je me répète pour i valant %d\n",i);  
}
```

par

```
for (i=0; i<15; i++)  
    printf ("Je me répète pour i valant %d\n",i);
```

### 9.3 Exercice 1

Utilisez une boucle pour `i` variant de 0 à 15 inclus pour afficher :

*"Ceci est la couleur i"*

où i est remplacé par sa valeur en position (1,i+1) avec la couleur i.

●\* cprintf ne supporte pas le \n permettant de retourner à la ligne; de ce fait utilisez i allant de 0 à 15 pour modifier le no de ligne.

#### *Aide*

```
gotoxy (1,i+1); /* +1 car le coin haut gauche est aux coordonnées (1,1) */
textcolor (i);
cprintf ("Ceci est la couleur %d",i);
```

#### **9.4 Notion de double boucle.**

Il est possible de remplacer les instructions par une boucle afin de réaliser une double boucle.

On obtient donc :

Pour i allant de ... à ...

```
{
    ...
    Pour j allant de ... à ...
    {
        ...
    }
}
```

#### *Exemple :*

```
#include <conio.h>

int main ()
{
    int i;
    int j;

    clrscr ();

    for (i=0; i<5; i++)
    {
        printf ("\nJe suis dans la boucle i, i vaut %d\n",i);

        for (j=3; j>0; j--)
        {
            printf ("Je suis dans la boucle j, j vaut %d\n",j);
        }
    }

    getch (); /* Attendre l'appui d'une touche */
}
```

#### **9.5 Exercice 2:**

En utilisant la double boucle, écrire un programme qui écrit 1 étoile, saute une ligne, écrit 2 étoiles, saute une ligne, écrit 3 étoiles ... jusqu'à 5 étoiles afin d'obtenir ceci :

```
*
**
***
```

\*\*\*\*  
\*\*\*\*\*

## 9.6 Exercice 3 : Et les shadoks fêtèrent Noël ...

### 9.6.1 "Cône" du sapin

A l'aide d'une double boucle et d'un gotoxy, réaliser un cône pour dessiner le haut du sapin en vert sur 12 lignes. Vous devriez obtenir ceci :

```
*  
***  
*****
```

sur 12 lignes ...

#### *Aide*

Sur la ligne n° 1, on affiche  $1 * 2 - 1 = 1$  étoile en position (42-1,1)  
Sur la ligne n° 2, on affiche  $2 * 2 - 1 = 3$  étoiles en position (42-2,2)  
Sur la ligne n° 3, on affiche  $3 * 2 - 1 = 5$  étoiles en position (42-3,3)

Sur la ligne n° i, on affiche  $(i * 2 - 1)$  étoile en position (42-i, i);

On obtient donc :

```
for (j=0; j<(i*2-1); j++)  
{  
    gotoxy (42-i+j,i);  
    printf ("*");  
}
```

### 9.6.2 Affichage du tronc

Pour poursuivre le sapin, il nous faut maintenant dessiner le tronc. Ecrire la suite du programme en dessinant le tronc à l'aide du caractère @. Vous devriez obtenir ceci (en brun) :

```
@ @ @  
@ @ @  
@ @ @
```

### 9.6.3 Affichage des boules de Noël

Afficher à l'aide de l'option blink des boules de Noël.

## 9.7 Exercice 4 : Table Ascii

Les codes Ascii (i.e. les nombres qui représentent les caractères en informatique) vont de 0 à 255. Ecrire un programme qui fait afficher à la suite (\n) puis de façon propre c'est à dire dans un tableau de 16 \* 16 les codes Ascii avec les caractères qui leur correspondent.

#### *Aide*

Pour faire afficher le caractère associé à un code Ascii, on écrit :

```
printf ("%d : %c", code_ascii, code_ascii);
```

*Exemple :*     int i = 65;  
              printf ("%d : %c", i, i);     affichera     65 : A

## Corrigés des exercices du chapitre 9

### ! **Exercice 1**

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

int main ()
{
    int i;
    clrscr ();

    for (i=0; i<=15; i++)
    {
        textbackground(15-i);
        gotoxy (1,i+1);
        textcolor (128+i);
        cprintf ("Ceci est la couleur %d", i);
    }

    getch ();
}
```

### ! **Exercice 2**

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

int main ()
{
    int i;
    int j;

    clrscr ();

    for (i=1; i<5; i++)
    {
        for (j=1; j<=i; j++)
        {
            textcolor (11);
            cprintf ("*");
        }
        printf ("\n");
    }

    getch ();
}
```

### ! **Exercice 3 : Et les shadoks fêtèrent Noël ...**

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

int main ()
{
    int i;
```

```
int j;

clrscr ();

for (i=1; i<=12; i++)
{
    for (j=1; j<= (i*2-1); j++)
    {
        gotoxy (41-i+j,i);
        textcolor (2);
        cprintf ("*");
    }

    printf ("\n");
}

for (i=1; i<=3; i++)
{
    for (j=3; j>=1; j--)
    {
        gotoxy (39+j,12+i);
        textcolor (6);
        cprintf ("@");
    }
}

textcolor (130);
gotoxy (41,3);
cprintf ("*");

getch ();
}
```

#### ! **Exercice 4 : Table Ascii**

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

void main ()
{
    int i;
    int j;

    clrscr ();

    for (i=0; i<255; i++)
    {
        printf ("%d -> %c\n",i,i);
    }

    getch ();
}
```

**Attention, certains code Ascii provoquent des problèmes d'affichage ...**

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

void main ()
{
    int i;
```

```
int j;  
int car = 0;  
  
clrscr ();  
  
printf (" 0  1  2  3  4  5  6  7  8  9  10 11 12 13 14 15\n");  
for (i=1; i<=16; i++)  
{  
    for (j=1; j<=16; j++)  
    {  
        printf ("%c  ",car);  
        car ++;  
    }  
    printf ("\n ");  
}  
getch ();  
}
```