

## 5 Remise en forme

### 5.1 Prologue

L'objet de ce cours est de réaliser un petit break dans l'apprentissage du C et de s'attacher à voir que l'on est capable de réaliser avec le peu de moyen que l'on a.

Ce cours sera donc constitué de 3 exercices de difficultés croissantes avec apprentissage d'une nouvelle fonction et d'un exercice complet de programmation.

L'exercice 4 est assez difficile et mérite surtout de bien réfléchir avant de se mettre au clavier.

### 5.2 Exercice 1.

Réaliser un programme qui saisisse un nombre et indique à l'utilisateur si celui-ci est plus grand ou plus petit qu'un autre nombre fixé par le programme.

*Exemple :*

```
si (nbre_saisi<10)  
alors "plus petit"
```

Reprendre l'exercice du chapitre 4 qui disait si un nombre est positif, négatif ou nul.

### 5.3 Retour sur Getch ()

La fonction `getch ()` permet d'attendre la frappe d'un caractère au clavier, de le lire et de le renvoyer. 2 utilisations peuvent être faites de `getch ()`, la première est celle permettant d'attendre la frappe d'une touche sans se soucier de sa valeur, la seconde est celle permettant de lire un caractère au clavier.

*Exemples:*

1. Attente  
`getch ();`
2. Saisie d'un caractère  
`char car;`  
`car = getch ();`

A chaque fois, `getch ()` effectue le même traitement :

- Attend la frappe d'une touche au clavier.
- Renvoie le caractère frappé.

Dans le 1<sup>er</sup> cas, ce caractère n'est simplement pas récupéré.

### 5.4 Boucle Faire ... Tant que (vrai)

Do ... while, traduisez par Faire Tant que permet de réaliser une suite d'événements tant qu'une condition ou un ensemble de conditions est rempli.

*Exemple*

```
char car;
```

```

int  sortie;

do
{
    clrscr ();
    printf ("Tapez S pour sortir ...");

    /* On saisit un caractère */
    car = getch ();

    /* On le compare pour savoir si l'on peut sortir */
    sortie = ((car == 's') || (car == 'S'));
} while (!sortie);

```

} Suite d'évènements

**Rappel :**

Un nombre entier vaut la valeur logique **vraie** si celui-ci est différent de 0.

Un nombre entier vaut la valeur logique **faux** si celui-ci est égal à 0.

|| signifie un ou logique (or).

**5.5 Exercice 2**

Tapez l'exemple précédent, aménagez le, comprenez le, puis transformez-le afin que l'on sorte de la boucle uniquement lorsque l'utilisateur a tapé le nombre 10.



**Attention** La saisie d'un nombre ne se fait pas par getch mais par scanf ...  
Cf. Chapitre 3.

**5.6 Exercice 3**

Voici un petit exemple de programme qui permet d'obtenir des nombres aléatoires entre 0 et 100.

Notes :

**random** et **randomize** sont définis dans la bibliothèque <stdlib.h>.

```

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

int main()
{
    int  nb_alea;    /* Nombre aléatoire */

    clrscr ();
    randomize ();

    /* Le nombre aléatoire est stocké dans une variable puis affiché */
    nb_alea = random (101);
    printf ("%d",nb_alea);

    /* Le nombre aléatoire est affiché directement à l'écran */
    printf ("%d",random (101));

    getch ();
}

```

**randomize** permet d'initialiser le système aléatoire.

**random** permet d'obtenir un nombre entre 0 et (n-1). [random (n)]

En vous aidant de ce petit programme et de ce qui a été fait précédemment, réaliser un petit jeu qui :

1. Initialise un nombre entre 0 et 100.
2. Tente de faire deviner ce nombre à l'utilisateur en lui indiquant s'il est plus petit ou plus grand.

Pour cet exercice, aidez vous de ce qui a été fait précédemment.

### 5.7 Exercice 4 : jeu de 421

Vous allez réaliser un mini jeu de 421. Les règles sont simples, vous avez 4 essais pour réaliser avec les 3 dés un jet donnant 421. Le programme lance les dés et affiche le résultat. Si c'est un 421, on affiche "Gagné", sinon on demande à l'utilisateur d'appuyer sur la touche R pour relancer. Au bout du 4<sup>ème</sup> essai, au lieu de proposer de relancer, on affichera "Perdu" et le programme se terminera.

#### 5.7.1 Analyse (ne pas toucher à l'ordinateur !).

Analyser l'énoncé du problème.

1. Identifier chaque étape du problème, ce qu'il faut faire pas à pas.
2. Analyser **chaque étape** en utilisant des mots *français* sans utiliser de terme informatique afin d'avoir un fonctionnement pas à pas ...

<b>Faites comme si vous étiez un robot auquel on apprend comment jouer.</b>
-----------------------------------------------------------------------------

#### 5.7.2 Programmation

Courage : Programmer le Jeu ...

## Corrigés des exercices du chapitre 5

### ! §5.2

```
#include <stdio.h>

int main ()
{
    int    nb_choisi = 33;
    int    nb_saisi = 0;

    clrscr (); /* Efface l'écran */

    printf ("Votre nombre : ");
    scanf ("%d",&nb_saisi);

    if (nb_choisi < nb_saisi)
        printf ("Mon nombre est plus petit");
    else
    {
        if (nb_choisi == nb_saisi)
            printf ("Mon nombre est égal");
        else
            printf ("Mon nombre est plus grand");
    }

    /* Attente */
    getch ();

    return (0);
}
```

### ! §5.4

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

int main ()
{
    int    valeur;

    do
    {
        clrscr ();
        printf ("Votre nombre : ");
        scanf ("%d",&valeur);
    }while (valeur != 10);
    return (0);
}
```

### ! §5.5

```
#include <stdio.h>
#include <stdlib.h> /* pour random */
#include <conio.h>

int main ()
```

```
{
    int  nb_hasard = 0;
    int  votre_nb = 0;

    randomize ();
    nb_hasard = random (101); /* Nombre entre 0 et 100 */

    do
    {
        clrscr ();

        printf("Saisie de votre nombre : ");
        scanf("%d",&votre_nb);

        if (nb_hasard < votre_nb)
        {
            printf ("\nMon nombre est plus petit");

            /* A cause du clrscr () qui nous empêcherait de voir le
            message */
            printf ("\nAppuyez sur une touche");
            getch ();
        }
        else
        {
            if (nb_hasard > votre_nb)
            {
                /* il peut être aussi égal ... */
                printf ("\nVotre nombre est plus grand");

                /* A cause du clrscr () qui nous empêcherait
                de voir le message */

                printf ("\nAppuyez sur une touche");
                getch ();
            }
        }
    }while (votre_nb != nb_hasard);

    printf ("\nTrouvé");
    getch ();

    return (0);
}
```

### ! **Le jeu du 421 : version longue**

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

/* Note : true ou vrai est équivalent à la valeur 1 */
/* false ou faux est équivalent à la valeur 0 */
main()
{
    int  de1, de2, de3;
    int  gagne;
    int  essai;

    essai = 0;
    gagne = 0;
```

```
randomize ();

do
{
    clrscr ();

    de1 = random (6) + 1;
    de2 = random (6) + 1;
    de3 = random (6) + 1;

    printf ("Lancer : [%d] [%d] [%d]",de1,de2,de3);
    getch ();

    if (de1 == 4)
    {
        if (de2 == 2)
        {
            if (de3 == 1)
                gagne = 1;
            else
                gagne = 0;
        }
        else
        {
            if (de2 == 1)
            {
                if (de3 == 2)
                    gagne = 1;
                else
                    gagne = 0;
            }
            else
                gagne = 0;
        }
    }

    if (de2 == 4)
    {
        if (de1 == 2)
        {
            if (de3 == 1)
                gagne = 1;
            else
                gagne = 0;
        }
        else
        {
            if (de1 == 1)
            {
                if (de3 == 2)
                    gagne = 1;
                else
                    gagne = 0;
            }
            else
                gagne = 0;
        }
    }

    if (de3 == 4)
```

```
{
    if (de2 == 2)
    {
        if (de1 == 1)
            gagne = 1;
        else
            gagne = 0;
    }
    else
    {
        if (de2 == 1)
        {
            if (de1 == 2)
                gagne = 1;
            else
                gagne = 0;
        }
        else
            gagne = 0;
    }
}

    essai = essai + 1;
}
while ((!gagne) && (essai != 4));

if (gagne)
    printf ("\nGagné !!!");
else
    printf ("\nPerdu ...");

getch ();
}
```

### ! **Le jeu du 421 : version courte**

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

```
main()
{
    int  de1, de2, de3;
    int  gagne;
    int  essai;

    essai = 0;

    randomize ();

    do
    {
        clrscr ();

        de1 = random (6) + 1;
        de2 = random (6) + 1;
        de3 = random (6) + 1;

        printf ("Lancer : [%d] [%d] [%d]", de1, de2, de3);
        getch ();

        gagne = ((de1 == 4) && (de2 == 2) && (de3 == 1));
    }
}
```

```
gagne = gagne || ((de1 == 4) && (de2 == 1) && (de3 == 2));
gagne = gagne || ((de1 == 2) && (de2 == 1) && (de3 == 4));
gagne = gagne || ((de1 == 2) && (de2 == 4) && (de3 == 1));
gagne = gagne || ((de1 == 1) && (de2 == 2) && (de3 == 4));
gagne = gagne || ((de1 == 1) && (de2 == 2) && (de3 == 4));

    essai = essai + 1;
}
while ((!gagne) && (essai != 4));

if (gagne)
    printf ("\nGagné !!!");
else
    printf ("\nPerdu ...");

getch ();

}
```