

Jeu des allumettes

Adaptée du livre de Jacques Deconchat : "102 Programmes Pour MO6, TO8 et TO9+"

Un certain nombre d'allumettes est disposé entre les deux partis, l'ordinateur et vous. Le but du jeu est de ne pas retirer la dernière allumette. Pour se faire, une prise maximale est désignée par le joueur.

En début de partie, on demande :

1. Le nombre d'allumettes disposé entre les deux joueurs (de 3 à 100).
2. Le nombre maximal d'allumettes que l'on peut retirer.
3. Qui commence (0 pour le joueur et 1 pour l'ordinateur).

Puis, tour à tour, chaque parti donne le nombre d'allumettes qu'il prend. L'ordinateur répond grâce à la fonction décrite ci-dessous :

```
int    jeu_ordi (int nb_allum, int prise_max)
{
    int prise = 0;
    int s = 0;
    float t = 0;

    s = prise_max + 1;
    t = ((float) (nb_allum - s)) / (prise_max + 1);

    while (t != floor (t))
    {
        s--;
        t = ((float) (nb_allum-s)) / (prise_max + 1);
    }

    prise = s - 1;
    if (prise == 0)
        prise = 1;

    return (prise);
}
```

La fonction floor donne l'arrondi inférieur d'un nombre.

Nb_allum est le nombre d'allumettes sur la table (au moment du coup à jouer).

Prise_max est le nombre maximum d'allumettes autorisé lors d'une prise.

La fonction retourne en sortie le nombre d'allumettes prises par l'ordinateur.

Exemple :

```
prise = jeu_ordi (10,3) ;
```

La partie se termine lorsqu'il n'y a plus d'allumettes sur la table. La personne ayant tiré la dernière allumette est le perdant, l'autre le vainqueur.

Exemple de solution

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

int jeu_ordi (int nb_allum, int prise_max)
{
    int prise = 0;
    int s = 0;
    float t = 0;

    s = prise_max + 1;
    t = ((float) (nb_allum - s)) / (prise_max + 1);

    while (t != floor (t))
    {
        s--;
        t = ((float) (nb_allum-s)) / (prise_max + 1);
    }

    prise = s - 1;
    if (prise == 0)
        prise = 1;

    return (prise);
}

int main ()
{
    int nb_max_d=0;          /*nbre d'allumettes maxi au d,part*/
    int nb_allu_max=0;       /*nbre d'allumettes maxi que l'on peut tirer
au maxi*/
    int qui=0;               /*qui joue? 0=Nous --- 1=PC*/
    int prise=0;             /*nbre d'allumettes prises par le joueur*/
    int nb_allu_rest=0;      /*nbre d'allumettes restantes*/

    clrscr ();

    printf ("Nombre d'allumettes disposées entre les deux joueurs (entre
1 et 300): \n");
    scanf ("%d",&nb_max_d);

    do
    {
        printf ("\nNombre maximal d'allumettes que l'on peut retirer:
\n");
        scanf ("%d",&nb_allu_max);
    }
    while (nb_allu_max >= nb_max_d);
    /* On répète la demande de prise tant que le nombre donné n'est pas
correct */

    printf ("\nQuel joueur commence? 0--> Joueur ; 1--> Ordinateur: \n");
    scanf ("%d",&qui);

    nb_allu_rest = nb_max_d;

    do
    {
```

```
printf ("\nNombre d'allumettes restantes : %d",nb_allu_rest);

if (qui==0)
{
    do
    {
        printf ("\nCombien d'allumettes souhaitez-vous
piocher?\n");
        scanf ("%d",&prise);
    }
    while ((prise > nb_allu_rest) || (prise > nb_allu_max));
    /* On répète la demande de prise tant que le nombre donné
n'est pas correct */
}
else
{
    prise = jeu_ordi (nb_allu_rest , nb_allu_max);
    printf ("\nPrise de l'ordi : %d\n",prise);
}

qui=!qui;

nb_allu_rest= nb_allu_rest - prise;
}
while (nb_allu_rest >0);

if (qui == 0)      /* C'est à nous de jouer */
{
    printf ("\nVous avez gagné!\n");
}
else
{
    printf ("\nVous avez perdu!\n");
}

getch ();

return (0);
}
```