

IN100 : Cours 11 – Le Morpion

Sandrine Vial
`sandrine.vial@uvsq.fr`

Décembre 2015

Tableaux à 2 dimensions

- ▶ Déclaration : `int T[10][20];` : un tableau contenant 10×20 entiers.
- ▶ Accès à un élément `T[i][j]`
- ▶ Passage de paramètres `void f(int T[10][])`
- ▶ Appel de la fonction `f` : `f(T);`

Le Morpion

- ▶ Programmer le jeu du morpion.
- ▶ Un humain joue contre l'ordinateur.
- ▶ Le jeu est modélisé dans un tableau T à 2 dimensions :
`int T[3][3];`

Etape 1 : écriture du main

```
#include "graphics.h"  
int jeu[3][3];  
int main()  
{
```

```
    exit(0);  
}
```

Etape 1 : écriture du main

```
#include "graphics.h"
int jeu[3][3];
int main()
{
    init_graphics(300,300);
```

```
    wait_escape();
    exit(0);
}
```

Etape 1 : écriture du main

```
#include "graphics.h"
int jeu[3][3];
int main()
{   int G;
    init_graphics(300,300);
    init_jeu();
    affiche_jeu();
    G = gagnant();
    while(!G)
    {

    }
    affiche_resultat();
    wait_escape();
    exit(0);
}
```

Etape 1 : écriture du main

```
#include "graphics.h"
int jeu[3][3];
int main()
{   int G;
    init_graphics(300,300);
    init_jeu();
    affiche_jeu();
    G = gagnant();
    while(!G)
    {
        jouer_humain();
        affiche_jeu();
        G = gagnant();
    }
    affiche_resultat();
    wait_escape();
    exit(0);
}
```

Etape 1 : écriture du main

```
#include "graphics.h"
int jeu[3][3];
int main()
{   int G;
    init_graphics(300,300);
    init_jeu();
    affiche_jeu();
    G = gagnant();
    while(!G)
    {
        jouer_humain();
        affiche_jeu();
        G = gagnant();
        if (!G)
        {
            jouer_ordinateur();
            affiche_jeu();
            G = gagnant();
        }
    }
    affiche_resultat();
    wait_escape();
    exit(0);
}
```


Etape 2 : Ecriture des fonctions "vides"

```
void init_jeu()  
{  
}
```

```
int gagnant()  
{  
    return 0;  
}
```

```
void jouer_humain()  
{  
}
```

```
void jouer_ordinateur()  
{  
}
```

```
void affiche_resultat()  
{  
}
```

```
void affiche_jeu()  
{  
}
```

Etape 3 : Ecriture de init_jeu

```
void init_jeu()
{
    int i,j;

    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++) jeu[i][j] = 0;
    }
}
```

Etape 4 : Ecriture de affiche_jeu

```
void affiche_jeu()
{
    fill_screen(black);
    affiche_quadrillage();
}
```

```
void affiche_quadrillage()
{
    POINT p1,p2;

    p1.x = 100;
    p1.y = 0;
    p2.x = p1.x;
    p2.y = 300;
    draw_line(p1,p2,blanc);
    p1.x = 200;
    p2.x = p1.x;
    draw_line(p1,p2,blanc);
    p1.x = 0;
    p1.y = 100;
    p2.x = 300;
    p2.y = p1.y;
    draw_line(p1,p2,blanc);
    p1.y = 200;
    p2.y = p1.y;
    draw_line(p1,p2,blanc);
}
```

Etape 5 : Ecriture de jouer_humain

```
void jouer_humain()
{
    POINT p;
    int i,j;

    do {
        p = wait_clic();
        i = p.x / 100;
        j = p.y / 100;
    } while(jeu[i][j] != 0);
    jeu[i][j] = 1;
}
```

Etape 6 : Modification de affiche_jeu

```
void affiche_jeu()
{
    int i,j;
    fill_screen(black);
    affiche_quadrillage();
    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            if(jeu[i][j] == 1)
                affiche_cercle(i,j);
        }
    }
}
```

Etape 6 : Modification de affiche_jeu

```
void affiche_jeu()
{
    int i,j;
    fill_screen(black);
    affiche_quadrillage();
    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            if(jeu[i][j] == 1)
                affiche_cercle(i,j);
        }
    }
}
```

```
void affiche_cercle(int i, int j)
{
    POINT centre;

    centre.x = i * 100 + 50;
    centre.y = j * 100 + 50;
    draw_circle(centre,20,rouge);
}
```

Etape 7 : Ecriture de jouer_ordinateur

```
void jouer_ordinateur()
{
    int i,j;
    int joue;
    joue = 0;
    i = 0;
    while (!joue && i < 3)
    {
        j = 0;
        while (!joue && j < 3)
        {
            if (jeu[i][j] == 0)
            {
                jeu[i][j] = 10;
                joue = 1;
            }
            j = j + 1;
        }
        i = i + 1;
    }
}
```

Etape 8 : Modification de affiche_jeu

```
void affiche_jeu()
{
    int i,j;
    fill_screen(black);
    affiche_quadrillage();
    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            if(jeu[i][j] == 1)
                affiche_cercle(i,j);
            else if(jeu[i][j]==10)
                affiche_croix(i,j);
        }
    }
}
```


Etape 8 : Modification de affiche_jeu

```
void affiche_jeu()
{
    int i,j;
    fill_screen(black);
    affiche_quadrillage();
    for(i=0;i<3;i++)
    {
        for(j=0;j<3;j++)
        {
            if(jeu[i][j] == 1)
                affiche_cercle(i,j);
            else if(jeu[i][j]==10)
                affiche_croix(i,j);
        }
    }
}
```

```
void affiche_croix(int i, int j)
{
    POINT p1,p2;
    int k;

    p1.x = i*100 + 30;
    p1.y = j*100 + 70;
    p2.x = i*100 + 70;
    p2.y = j*100 + 30;
    draw_line(p1,p2,rouge);
    k = p1.y;
    p1.y = p2.y;
    p2.y = k;
    draw_line(p1,p2,rouge);
}
```

Etape 9 : Ecriture de gagnant

```
int gagnant()
{
    int i,somme = 0;
    int gagnant = 0;
    i = 0;
    while (!gagnant && i < 3)
    {
        somme = somme_ligne(i);
        if (somme == 3 || somme == 30) gagnant = 1;
        else
        {
            somme = somme_colonne(i);
            if (somme == 3 || somme == 30) gagnant = 1;
        }
        i = i + 1;
    }
    if (gagnant) return somme;
    somme = somme_diagonale1();
    if (somme == 3 || somme == 30) return somme;
    somme = somme_diagonale2();
    if (somme == 3 || somme == 30) return somme;

    return 0;
}
```

Etape 10 : Ecriture des somme_...

```
int somme_ligne(int i)
{
    int j,s;

    s = 0;
    for(j=0;j<3;j++)
        s = s + jeu[i][j];
    return s;
}
```

```
int somme_colonne(int i)
{
    int j,s;

    s = 0;
    for(j=0;j<3;j++)
        s = s + jeu[j][i];
    return s;
}
```

```
int somme_diagonale1()
{
    int s,i;
    s = 0;
    for(i=0;i<3;i++)
        s = s + jeu[i][i];

    return s;
}
```

```
int somme_diagonale2()
{
    int s,i;
    s = 0;
    for(i=0;i<3;i++)
        s = s + jeu[2-i][i];

    return s;
}
```

Etape 11 : Ecriture de affiche_resultat

```
void affiche_resultat(int G)
{
    if (G == 3) write_text("L'humain gagne!");
    else if (G == 30) write_text("L'ordinateur gagne!");
    else write_text("Match nul!");
    writeln();
}
```

Etape 12 : modification du main

```
int main()
{ int G; int nb = 0;
  init_graphics(300,300);
  init_jeu();
  affiche_jeu();
  G = gagnant();
  while(!G && nb != 9)
  {
    jouer_humain();
    affiche_jeu();
    nb = nb + 1;
    G = gagnant();
    if (!G)
    {
      jouer_ordinateur();
      affiche_jeu();
      nb = nb + 1;
      G = gagnant();
    }
  }
  affiche_resultat(G);
  wait_escape();
  exit(0);
}
```