

09/12/2021

Compte rendu n°4 :

Projet 4A – ILC :
Développement Pac-Man en C++

Déclaration de la classe 'Ghost' qui comme la classe 'Hero' hérite de 'BaseCharacter'. Nous allons donc définir comment dessiner l'allure général d'un fantôme et voir pour éditer la couleur et donc créer plusieurs fantômes de couleurs différentes.

```
1  #ifndef GHOST_H
2  #define GHOST_H
3
4  #include "BaseCharacter.h"
5
6  /*
7   Couleur et nom des fantomes:
8   - Shadow (Ombre) - Blinky : le fantôme rouge.
9   - Speedy (Rapide) - Pinky : le fantôme rose.
10  - Bashful (Timide) - Inky (ou encore Inkey) : le fantôme bleu.
11  - Pokey (Limité) - Clyde : le fantôme orange.
12  */
13
14  class Ghost : public BaseCharacter
15  {
16  public:
17      // Constructeur
18      Ghost() : BaseCharacter() {};
19
20      void Draw(); //méthode hérité de BaseCharacter
21      void DrawC(int color);
22
23  };
24  #endif
```

Figure 1 : Ghost.h

On hérite de la méthode 'Draw()' qui ne nous servira pas dans cette classe car nous voulons ajouter en paramètre la couleur souhaitée pour nos fantômes, ajouté ici à la méthode 'DrawC()'.

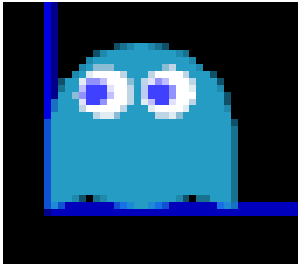
```

1  #include "Ghost.h"
2  #include "Hero.h"
3
4  extern Hero pacman;
5  extern void LabyDraw();
6
7
8  void Ghost::Draw(){}
9
10 void Ghost::DrawC(int color)
11 {
12     glPushMatrix();
13     int couleur = color;
14
15     switch (couleur) {
16         case 1://rouge
17             std::printf("1");
18             glColor3d(0.79, 0.14, 0.05); //valeur RGB/255 car prend des valeurs de 0 à 1
19             break;
20         case 2://rose
21             std::printf("4");
22             glColor3d(0.85, .6, .71);
23             break;
24         case 3://bleu
25             std::printf("2");
26             glColor3d(0.14, .61, .77);
27             break;
28         case 4://orange
29             std::printf("3");
30             glColor3d(0.86, .55, .02);
31             break;
32     }
33     //glColor3d(0.79, 0.14, 0.05); //rouge
34     glTranslated(PosC + 0.5, PosL + 0.5, 0.0); // Définition du centre de notre zone de dessin
35
36     // Sphère de la tête
37     glutSolidSphere(0.5, 12, 12);
38
39     // Carre du corps
40     glBegin(GL_QUADS);
41     glVertex3f(-0.5, 0.0, 0.0);
42     glVertex3f(-0.5, 0.4, 0.0);
43     glVertex3f(0.5, 0.4, 0.0);
44     glVertex3f(0.5, 0.0, 0.0);
45     glEnd();
46
47     // Bas du corps
48     // Triangle 1
49     glBegin(GL_TRIANGLES);
50     glVertex3f(-0.5, 0.4, 0);
51     glVertex3f(-0.25, 0.4, 0);
52     glVertex3f(-0.5, 0.5, 0);
53     glEnd();
54     // Triangle 2
55     glBegin(GL_TRIANGLES);
56     glVertex2f(-0.25, 0.4);
57     glVertex2f(0, 0.4);
58     glVertex2f(0, 0.5);
59     glEnd();
60     // Triangle 3
61     glBegin(GL_TRIANGLES);
62     glVertex2f(0, 0.4);
63     glVertex2f(0.25, 0.4);
64     glVertex2f(0, 0.5);
65     glEnd();
66     // Triangle 4
67     glBegin(GL_TRIANGLES);
68     glVertex2f(0.25, 0.4);
69     glVertex2f(0.5, 0.4);
70     glVertex2f(0.5, 0.5);
71     glEnd();
72
73     // Yeux
74     glColor3d(1, 1, 1); //Couleur blanche
75     // Oeil gauche // il y certaines bizarteries avec la creation de cercles
76     glTranslated(-0.2, -0.2, 0.0);
77     glutSolidSphere(0.15, 12, 12);
78
79     glColor3d(0.25, 0.25, 1);
80     glTranslated(-0.05, 0, 0.0);
81     glutSolidSphere(0.08, 12, 12);
82     // Oeil droit
83     glColor3d(1, 1, 1); //Couleur blanche
84     glTranslated(0.4, 0.0, 0.0);
85     glutSolidSphere(0.15, 12, 12);
86     glColor3d(0.25, 0.25, 1);
87     glTranslated(-0.05, 0, 0.0);
88     glutSolidSphere(0.08, 12, 12);
89
90     glPopMatrix();
91 }

```

Figure 2 : Ghost.cpp

On a au début de la déclaration de 'DrawC()' un switch case qui en fonction de la valeur mise en paramètre, permet de modifier la couleur générale de l'objet.



Pour dessiner notre fantôme, comme pour Pac-Man la dernière fois, nous allons faire cela en plusieurs fois. On commence par le carré du corps puis la sphère de la tête. Pour recréer le bas du corps du fantôme, nous allons ajouter des carrés noirs par-dessus l'objet et ainsi masquer certaines parties du corps. Pour finir on ajoute des yeux au fantôme en faisant un cercle blanc puis un cercle bleu. Voici un fantôme après compilation.

```
Hero pacman;
/*
Couleur et nom des fantomes:
- Shadow (Ombre) - Blinky : le fantôme rouge.
- Speedy (Rapide) - Pinky : le fantôme rose.
- Bashful (Timide) - Inky (ou encore Inkey) : le fantôme bleu.
- Pokey (Limité) - Clyde : le fantôme orange.
*/
Ghost blinky;
Ghost pinky;
Ghost inky;
Ghost clyde;
```

Figure 3 : Déclaration des personnages

Pour pouvoir placer nos personnages, j'ajoute un 'Switch case' qui va parcourir notre matrice et lorsqu'il va rencontrer un caractère spécifique, va placer notre objet préalablement défini en définissant ses coordonnées. Pour que cela fonctionne j'ai donc attribué une lettre à chaque personnage dans le fichier texte et dans le switch case :

- H pour Hero ; notre Pac-Man
- B pour Blinky ; fantôme rouge
- P pour Pinky ; fantôme rose
- I pour Inky ; fantôme bleu
- C pour Clyde ; fantôme orange

```
119 // Lecture du tableau
120 for (int j = 0; j < NbL; j++)
121     for (int i = 0; i < NbC; i++)
122     {
123         lvlFile >> Matrix[i][j];
124         switch (Matrix[i][j])
125         {
126             case 'h':
127             case 'H':
128                 // Définir ici la position de départ
129                 pacman.SetPosC(i);
130                 pacman.SetPosL(j);
131                 break;
132             case 'b':
133             case 'B':
134                 blinky.SetPosC(i);
135                 blinky.SetPosL(j);
136                 break;
137             case 'p':
138             case 'P':
139                 pinky.SetPosC(i);
140                 pinky.SetPosL(j);
141                 break;
142             case 'i':
143             case 'I':
144                 inky.SetPosC(i);
145                 inky.SetPosL(j);
146                 break;
147             case 'c':
148             case 'C':
149                 clyde.SetPosC(i);
150                 clyde.SetPosL(j);
151                 break;
152         }
153     }
```

Figure 4 : Placement des personnages depuis fichier texte

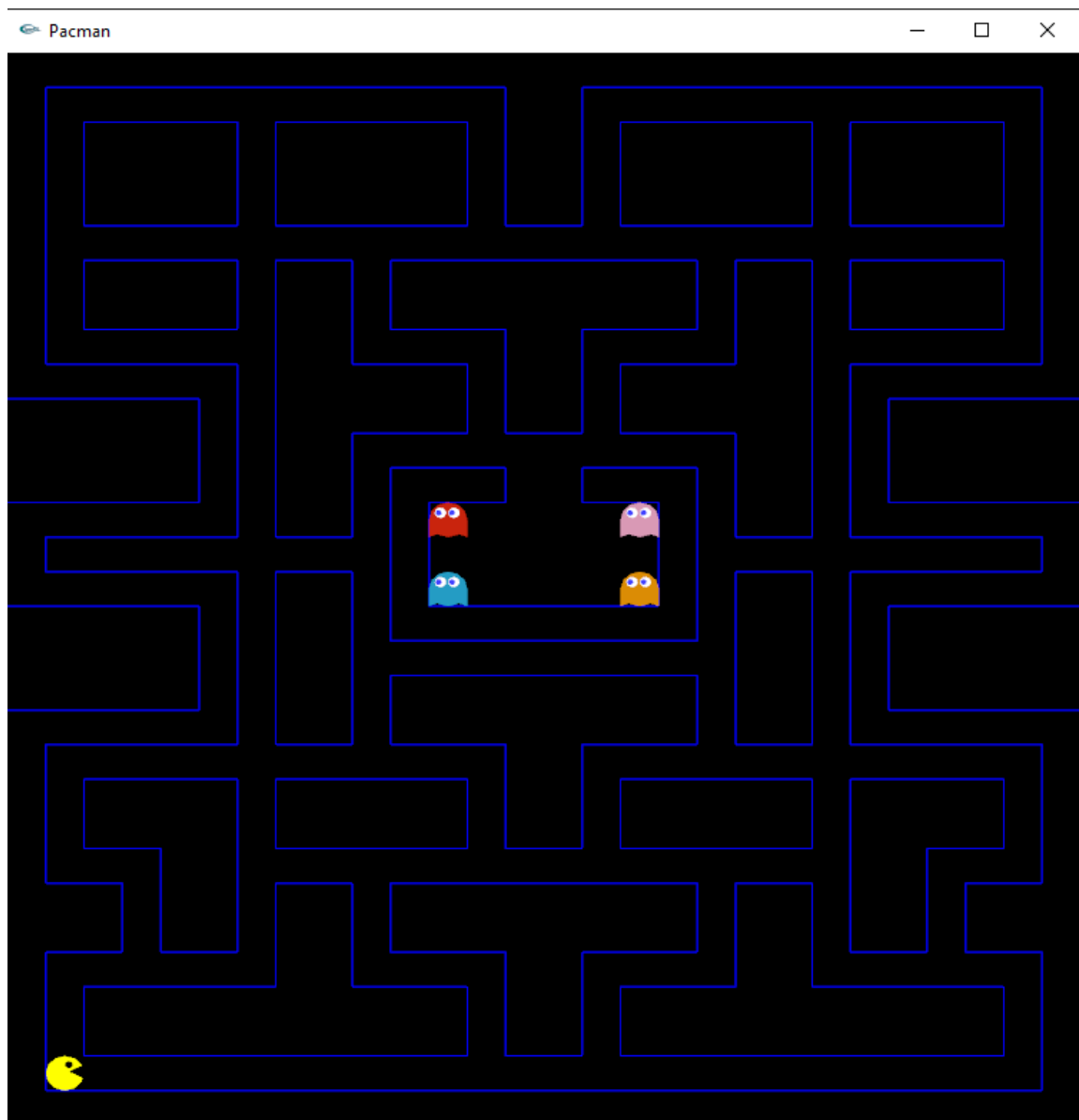


Figure 5 : Résultat du rendu après compilation