

Introduction à la Programmation

TP #2 - Corrigé

G1: monasse(at)imagine.enpc.fr G2: nicolas.audebert(at)onera.fr
G3: alexandre.boulch(at)onera.fr G4: maxime.ferrera(at)onera.fr
G5: laurent.bulteau(at)u-pem.fr G6: pierre-alain.langlois(at)eleves.enpc.fr

1 Corrigé

1.1 Solution

1.1.1 Projet hop

En notant ? les variables non initialisées, [i] un "lien"¹ vers i, en indiquant les variables par leur fonction, et en laissant la case vide quand la variable n'existe pas, on obtient le tableau d'exécution suivant :

Ligne	<i>i</i> _{main}	<i>j</i> _{main}	<i>k</i> _{main}	<i>a</i> _{add}	<i>b</i> _{add}	<i>c</i> _{add}	<i>ret</i> _{add}	<i>a</i> _{triple1}	<i>a</i> _{triple2}
23	?	2	?						
24	3	2	?						
25	3	2	?						
6	3	2	?	3	2				
7	3	2	?	3	2	5			
8	3	2	?	3	2	5			
25	3	2	?				5		
25	3	2	5						
26	3	2	5						
13	3	2	5					3	
14	3	2	5					9	
27	3	2	5						
18	3	2	5						[<i>i</i> _{main}]
19	9	2	5						[<i>i</i> _{main}]
28	9	2	5						

1.1.2 Projet Tennis

Voici le source de la solution :

```
1 // Imagine++ project
2 // Project: Tennis
3 // Author: renaud
4 // Date: 2007/10/9
5 // TP #2 du cours de C++ de l'ENPC
6
7 #include <Imagine/Graphics.h>
8 #include <iostream>
9 using namespace Imagine;
10 using namespace std;
11
12 ///////////////////////////////////////////////////////////////////
13 // Constantes du programme
14 const int width=256; // Largeur de la fenetre
15 const int height=256; // Hauteur de la fenetre
16 const int ball_size=3; // Rayon de la balle
```

1. = référence.

```

17 const int raq_width=20; // Demi-largeur des raquettes
18 const int raq_height=3; // Demi-hauteur des raquettes
19 const int raq_speed=4; // Vitesse des raquettes
20
21 ///////////////////////////////////////////////////////////////////
22 // Fonction de gestion du clavier
23 // Pour l'instant, utiliser sans se soucier du fonctionnement
24 // Retourne dans sens1 (resp. sens2) -1, 1 ou 0 pour le joueur 1
25 // (resp. joueur 2) suivant qu'il appuie sur la touche gauche,
26 // droite ou sur aucune touche.
27 void Clavier(int& sens1, int& sens2) {
28     static bool left1=false, right1=false, left2=false, right2=false;
29     Event e;
30     do {
31         getEvent(0, e);
32         if (e.type==EVT_KEY_ON || e.type==EVT_KEY_OFF) {
33             bool push=(e.type==EVT_KEY_ON);
34             switch (e.key) {
35                 case 'S':
36                     left1=push;
37                     break;
38                 case 'D':
39                     right1=push;
40                     break;
41                 case 'K':
42                     left2=push;
43                     break;
44                 case 'L':
45                     right2=push;
46                     break;
47             }
48         }
49     } while (e.type!=EVT_NONE);
50     sens1=(left1?-1:(right1?1:0));
51     sens2=(left2?-1:(right2?1:0));
52 }
53
54 ///////////////////////////////////////////////////////////////////
55 // Affichage d'une balle
56 void DessineBalle(int x, int y, Color col) {
57     fillRect(x-ball_size, y-ball_size, 2*ball_size+1, 2*ball_size+1, col);
58 }
59
60 ///////////////////////////////////////////////////////////////////
61 // Affichage d'une raquette
62 void DessineRaquette(int x, int y, Color col) {
63     fillRect(x-raq_width, y-raq_height, 2*raq_width+1, 2*raq_height+1, col);
64 }
65
66 ///////////////////////////////////////////////////////////////////
67 // Deplacement d'une balle
68 void BougeBalle(int &x, int &y, int &u, int &v, int &score1, int &score2) {
69     // Rebond sur les bords gauche et droit
70     if (x+u>width-ball_size || x+u<ball_size)
71         u=-u;
72     // Rebond sur les bords haut et bas et comptage du score
73     if (y+v<ball_size) {
74         v=-v;
75         score1++;
76         cout << score1 << "-" << score2 << endl;
77     }

```

```

78     else if (y+v>height-ball_size) {
79         v=-v;
80         score2++;
81         cout << score1 << "-" << score2 << endl;
82     }
83     // Mise a jour de la position
84     x+=u;
85     y+=v;
86 }
87
88 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
89 // Deplacement d'une raquette
90 void BougeRaquette(int &x, int sens) {
91     if (sens==-1) {
92         x-=raq_speed;
93         if (x<raq_width)
94             x=raq_width;
95     } else if (sens==1) {
96         x+=raq_speed;
97         if (x>width-raq_width)
98             x=width-raq_width;
99     }
100 }
101
102 ////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////////
103 // Fonction principale
104 int main()
105 {
106     // Ouverture de la fenetre
107     openWindow(width,height);
108     // Position de la raquette 1
109     int xr1=width/2,
110         yr1=height-10;
111     // Position de la raquette 2
112     int xr2=width/2,
113         yr2=10;
114     // Position et vitesse de la balle
115     int xb=xr2,
116         yb=yr2+raq_height+ball_size,
117         ub=2,
118         vb=3;
119     // Scores des deux joueurs
120     int score1=0,
121         score2=0;
122     // Boucle principale
123     while (true) {
124         // Affichage de la balle, des raquettes et du score
125         noRefreshBegin();
126         DessineBalle(xb,yb,RED);
127         DessineRaquette(xr1,yr1,BLUE);
128         DessineRaquette(xr2,yr2,BLUE);
129         noRefreshEnd();
130         // Temporisation
131         milliSleep(20);
132         // Rebond sur la raquette 1
133         if (yb<=yr1-ball_size-raq_height &&
134             yb+vb>yr1-ball_size-raq_height &&
135             abs(xb-xr1)<=raq_width)
136             vb=-vb;
137         // Rebond sur la raquette 2
138         if (yb>=yr2+ball_size+raq_height &&

```

```

139         yb+vb<yr2+ball_size+raq_height &&
140         abs(xb-xr2)<=raq_width)
141         vb=-vb;
142     // Gestion du clavier
143     int sens1 ,sens2;
144     Clavier(sens1 ,sens2);
145     // Effacement de la balle , des raquettes et du score
146     noRefreshBegin();
147     DessineBalle(xb ,yb ,WHITE);
148     DessineRaquette(xr1 ,yr1 ,WHITE);
149     DessineRaquette(xr2 ,yr2 ,WHITE);
150     noRefreshEnd();
151     // Deplacement de la balle
152     BougeBalle(xb ,yb ,ub ,vb ,score1 ,score2);
153     // Deplacement des raquettes
154     BougeRaquette(xr1 ,sens1);
155     BougeRaquette(xr2 ,sens2);
156 }
157 endGraphics();
158 return 0;
159 }

```