

## Exercices d'informatique - Corrigé de la série n° 2 bis

Cours 3INOC01

### 1. Jeu du Canon.

On peut, par exemple, utiliser le script suivant:

```
1 from turtle import *
2 from math import *
3 import random
4 title("Jeu du canon")
5 setup (width=1250, height=850)
6 def tir(x,a,v0,m,cible):
7     def intersection(a0,b0,a1,b1):
8         m=[[b1[0]-a1[0],a0[0]-b0[0]],[b1[1]-a1[1],a0[1]-b0[1]]]
9         z=[a0[0]-a1[0],a0[1]-a1[1]]
10        det=m[0][0]*m[1][1]-m[1][0]*m[0][1]
11        if det==0:
12            return False
13        else:
14            mu=float(m[1][1]*z[0]-m[0][1]*z[1])/det
15            nu=float(-m[1][0]*z[0]+m[0][0]*z[1])/det
16            if mu<0 or mu>1 or nu<0 or nu>1:
17                return False
18            else:
19                return [a1[0]+mu*(b1[0]-a1[0]),a1[1]+mu*(b1[1]-a1[1])]
20    points=0
21    N0=100
22    goto(x+5,-350)
23    down()
24    width(1)
25    color('red')
26    cx0=x+5
27    cy0=-350
28    n=0
29    while n<=N0:
30        t=float(n)/N0*cos(a)*v0/5
31        cx=x+5-t*sin(a)*v0
32        cy=-5*t**2+cos(a)*v0*t-350
33        z=(300-cx)*(cx+100)
34        z0=(300-cx0)*(cx0+100)
35        if z>=0 or z0>=0:
36            n1=int((float(cx)+100)/4)
37            n2=min(int((float(cx0)+100)/4),99)
38            for k in range(n2,n1-1,-1):
39                a1=m[k]
40                b1=m[k+1]
41                i=intersection([cx0,cy0],[cx,cy],a1,b1)
42                if i!=False:
43                    cx=i[0]
44                    cy=i[1]
45                    points=-200
46                    n=N0
47                    break
48            goto(cx,cy)
49            cx0=cx
50            cy0=cy
51            n=n+1
52    up()
53    if cy<-349:
54        if cx<=cible+20 and cx>=cible:
55            points=200#touche la cible
56        elif cx<=-350:
```

```

57         points=-100#tombe dans l'eau
58     elif cx>300:
59         points=-80#retombe avant la montagne
60     else:
61         points=-50#ne touche pas la cible et ne tombe pas dans l'eau
62     write(str(points)+'Points')
63     return points
64 def canon(x,a):
65     c=[[0,20],[5,20],[5,0]]
66     m=0
67     for couleur in ['white','red']:
68         up()
69         color(couleur)
70         goto(x[m],-349)
71         down()
72         fill(1)
73         for n in range(0,3,1):
74             cx=cos(a[m])*c[n][0]-sin(a[m])*c[n][1]
75             cy=sin(a[m])*c[n][0]+cos(a[m])*c[n][1]
76             goto(x[m]+cx,-349+cy)
77         m=m+1
78         fill(0)
79         up()
80 def cible(x):
81     c=[[0,20],[20,20],[20,0]]
82     m=0
83     for couleur in ['white','blue']:
84         color(couleur)
85         goto(x[m],-349)
86         down()
87         fill(1)
88         for n in range(0,3,1):
89             cx=c[n][0]+x[m]
90             cy=c[n][1]-349
91             goto(cx,cy)
92         m=m+1
93         fill(0)
94         up()
95 def montagne():
96     N=100
97     g=lambda x: -x**2/100+50.
98     m=[]
99     up()
100    #lac
101    goto(-600,-350)
102    down()
103    width(3)
104    fill(1)
105    goto(-600,-400)
106    goto(-400,-400)
107    goto(-350,-350)
108    color('light blue')
109    fill(0)
110    color('black')
111    #montagne
112    goto(-100,-350)
113    fill(1)
114    for n in xrange(0,N+1,1):
115        x=-100+(400./N)*n
116        a=int(200*float(n*(n-N))/N**2)
117        y2=random.choice([-a,a+1,1])
118        y=g(x-100)+y2
119        m.append([x,y])
120        goto(x,y)
121    goto(300,-350)

```

```
122     color('green')
123     fill(0)
124     color('black')
125     goto(600,-350)
126     width(1)
127     goto(600,400)
128     goto(-600,400)
129     goto(-600,-350)
130     up()
131     return m
132 m=montagne()
133 xcanon=350
134 xcible=-350
135 a0=pi/4
136 a=a0
137 points=0
138 v0=100
139 while True:
140     x0canon=xcanon
141     x0cible=xcible
142     xcanon=350+random.getrandbits(10)*200/1024.
143     xcible=-350+random.getrandbits(10)*200/1024.
144     canon([x0canon,xcanon],[a0,a])
145     cible([x0cible,xcible])
146     for j in range(0,7,1):
147         a0=a
148         while True:
149             v0=raw_input('(vitesse initiale >0) v0=')
150             try:
151                 v0=float(v0)
152                 if v0>0:
153                     break
154             except:
155                 print "Ce n'est pas un nombre !"
156         while True:
157             a=raw_input('(angle entre 0 et 90) a=')
158             try:
159                 a=float(a)
160                 if a>=0 and a<=90:
161                     break
162             except:
163                 print "Ce n'est pas un nombre !"
164
165     a=(pi/2)*(1-a/90.)
166     x0canon=xcanon
167     canon([x0canon,xcanon],[a0,a])
168     p=tir(xcanon,a,v0,m,xcible)
169     points=points+p
170     print 'Total: '+str(points)+' Points'
```

---