

# Les nageurs - Corrigé

## NUMWORKS

Cet exercice est adapté de l'activité proposée par [Eric Schrafstetter](#).

### 1 Dessiner les nageurs

#### 1.1 Introduction

Définir la fonction `draw1` qui dessine la première ligne de la croix.

```
from kandinsky import*
from random import*
black = color(0,0,0)
white = color(255,255,255)

L1=[1,0,0,0,1]
L2=[0,1,0,1,0]
L3=[0,0,1,0,0]
L4=[0,1,0,1,0]
L5=[1,0,0,0,1]

def draw1(x,y):
    for i in range (5):
        if L1[i]==1:
            set_pixel(x+i,y,black)
        else:
            set_pixel(x+i,y,white)
```

Définir la fonction `draw_cross` qui dessine la croix aux coordonnées `(x,y)`.

```
from kandinsky import*
from random import*
black = color(0,0,0)
white = color(255,255,255)

L1=[1,0,0,0,1]
L2=[0,1,0,1,0]
L3=[0,0,1,0,0]
L4=[0,1,0,1,0]
L5=[1,0,0,0,1]
L=[L1,L2,L3,L4,L5]

def draw_cross(x,y):
    for i in range (5):
        for j in range (5):
            if L[j][i]==1:
                set_pixel(x+i,y+j,black)
            else:
                set_pixel(x+i,y+j,white)
```



FIGURE 1 – croix



FIGURE 2 – nageur

## 1.2 Dessin du nageur

Définir la fonction `draw_swimmer` qui dessine le nageur en position initiale.

```

from kandinsky import*
from random import*
black = color(0,0,0)
white = color(255,255,255)

p1=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
]

def draw_swimmer(x,y,p):
    for i in range (16):
        for j in range (16):
            if p[j][i]==1:
                set_pixel(x+i,y+j,black)
            else:
                set_pixel(x+i,y+j,white)

```

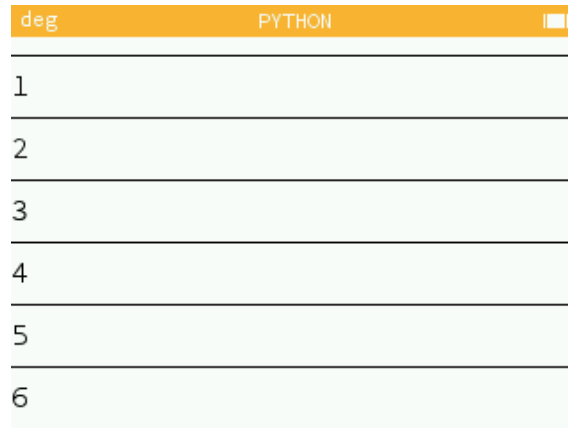


FIGURE 3 – piscine

### 1.3 Autres positions du nageur

Voici les listes p1, p2, p3 et p4.

```
p1=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
p2=[[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
p3=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,1,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
p4=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,1,1,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
```

## 2 Dessin de la piscine

```
from kandinsky import*
from random import*
black = color(0,0,0)
white = color(255,255,255)

def draw_pool():
    for i in range (6):
        draw_string(str(i+1),0,19+35*i)
        for j in range (320):
            set_pixel(j,35*i+10,black)
```

## 3 Animation de la course aléatoire des nageurs

```
from kandinsky import*
from random import*
black = color(0,0,0)
white = color(255,255,255)

p1=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
p2=[[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
p3=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,1,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,1,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,1,0],
p4=[[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],[0,0,0,0,0,0,0,0,1,1,0,0,0,0,0],[0,0,0,0,0,0,0,0,0,0,0,0,0,0,0],
```







deg	PYTHON	
1		
2		
3		
4		
5		
6		

FIGURE 4 – départ de la course

```

nage=[p1,p2,p3,p4]

def draw_swimmer(x,y,pos):
    for i in range (16):
        for j in range (16):
            if pos[j][i]==1:
                set_pixel(x+i,y+j,black)
            else:
                set_pixel(x+i,y+j,white)

def draw_pool():
    for i in range (6):
        draw_string(str(i+1),0,19+35*i)
        for j in range (320):
            set_pixel(j,35*i+10,black)

def swim():
    draw_pool()
    pos=[300,300,300,300,300,300]
    rang=0
    for j in range (6):
        draw_swimmer(300,j*35+20,p1)
    while max(pos)>10:
        i=randint(0,5)
        if pos[i]>10:
            draw_swimmer(pos[i]-1,i*35+20,nage[pos[i]%4])
            pos[i]=pos[i]-1
            if pos[i]==10:
                rang=rang+1
            draw_string(str(rang),300,19+35*i)

```