```
import pyxel

# taille de la fenetre 128x128 pixels
taille=128
pyxel.init(taille, taille, title="Infinity jump")
liste_PF=[[20,100],[70,65],[25,25],[76,-5],[20,-40],[50,-70],[76,-105],[20,-140],[60,-180],[25,-220],[70,-250],[25,-285],[76,-325],[20,-360],[70,-390],[25,-425],[70,-450],[25,-485],[76,-515],[20,-545],[50,-575],[76,-605],[20,-640],[60,-680],[25,-720],[70,-750],[25,-785],[76,-825],[20,-860],[70,-890],[25,-925],[70,-960],[25,-1000],[76,-1030],[20,-1065],[50,-1100],[76,-1130],[20,-1165],[60,-1195],[25,-1225],[70,-1255],[25,-1290],[76,-1330],[20,-1365],[70,-1395],[25,-1430],[70,-1455],[25,-1495],[76,-1525],[20,-1555],[50,-1585],[76,-1620],[20,-1655],[60,-1695],[25,-1735],[70,-1765],[25,-1800],[76,-1840],[20,-1875],[70,-1905],[25,-1945]]
perso_x=liste_PF[0][0]+16
perso_y=liste_PF[0][1]-8
perso_x0 = 0
perso_y0 = 0
x0=0
y0=0
repos = True
i=1
gagné=True
score = 0
longueur=32
#chargement du gamePack
pyxel.load("14.pyxres")

def draw():
    global perso_x, perso_y, gagné, repos
    pyxel.cls(0)
    #Blitter les murs
    pyxel.blt(0,0,1,6,0,20,taille,2)
    pyxel.blt(taille-20,0,1,48,0,20,taille,2)
    #Blitter la cascade
    pyxel.blt(20,6,2,14,0,88,116,2)
```

```
35      #Blitter le personnage en fonction de son orientation
36      if i == 1:
37          pyxel.blt(perso_x, perso_y,0,0,16,8,8,2)
38      if i == 2:
39          pyxel.blt(perso_x, perso_y,0,0,64,8,8,2)
40      #Plateformes
41      for plateforme in liste_PF:
42          pyxel.blt(plateforme[0], plateforme[1],0,0,72,longueur,5,2)
43      #Score
44      pyxel.text(102,8,str(score),9)
45      #Gagner
46      if (liste_PF[-1][0]-4<perso_x<liste_PF[-1][0]+28) and (liste_PF[-1][1]-9<perso_y<liste_PF[-1][1]-7):
47          repos=True
48          gagné=False
49          pyxel.blt(36,6,0,0,80,58,13)
50      #Perdre
51      if perso_y>taille:
52          repos=True
53          gagné=False
54          pyxel.text(50,20,"Perdu !",9)
55      #Code triche
56      if pyxel.btn(pyxel.KEY_A) and gagné==False:
57          gagné=True
58          perso_x=liste_PF[score][0]+16
59          perso_y=liste_PF[score][1]-8
60
61  def deplacer_perso(x, y):
62      global perso_y, perso_x0, perso_y0, repos, i, score
63      #déplacement avec les touches de directions
64      d = 3
65      if gagné and repos:
66          if pyxel.btn(pyxel.KEY_RIGHT) and perso_x < taille-29 :
```

```
67              x = x + d
68              i = 1
69          if pyxel.btn(pyxel.KEY_LEFT) and perso_x > 21 :
70              x = x - d
71              i = 2
72          for plateforme in liste_PF:
73              if plateforme[0]-4<perso_x<plateforme[0]+28 and 3<plateforme[1]-perso_y<=8 and
74  perso_x>=20 and perso_x<=taille-28:
75                  if pyxel.btnr(pyxel.KEY_SPACE):
76                      #declenche le saut
77                      perso_y=perso_y-3
78                      perso_x0=perso_x
79                      perso_y0=perso_y
80                      repos = not repos
81                      score = score + 1
82      return x,y
83
84  def sauter():
85      global perso_x, perso_y
86      if not repos :
87          dx = perso_x - perso_x0
88          #Paramètres parabole
89          a = .03
90          b = -1.6
91          c = perso_y0
92          if i == 1 :
93              perso_y = a*dx**2+b*dx+c
94              perso_x = perso_x + 1 #translation rectiligne uniforme selon x
95              tester_collision()
96          if i == 2 :
97              perso_y = a*(-dx)**2+b*(-dx)+c
98              perso_x = perso_x - 1
```

```
 99             tester_collision()
100
101     def tester_collision():
102         global perso_y, repos
103         for plateforme in liste_PF:
104             #Après un saut
105             if plateforme[0]-4<perso_x<plateforme[0]+28 and 3<plateforme[1]-perso_y<=8:
106                 repos = True
107                 perso_y=plateforme[1]-8
108                 perso_y=perso_y+.5
109                 break
110             #Sortie de plateforme et tomber avec la plateforme
111             if repos and (perso_x<=plateforme[0]-4 or perso_x>=plateforme[0]+28):
112                 perso_y=perso_y+.1
113         #Contre un mur
114         if not repos and (perso_x+8>=taille-20 or perso_x<=20):
115             repos = True
116
117     def deplacer_plateformes():
118         if gagné:
119             for plateforme in liste_PF:
120                 plateforme[1]=plateforme[1]+.5
121
122     def update():
123         global perso_x, perso_y
124         perso_x, perso_y = deplacer_perso(perso_x, perso_y)
125         sauter()
126         tester_collision()
127         deplacer_plateformes()
128
129     pyxel.run(update, draw)
```