

Deber Seminario  
Fausto Fabián Crespo Fernández  
Gato de Arnold

Se generó un punto inicial aleatorio y se itero un número suficiente de veces el mapa del gato de Arnold hasta que el punto de la órbita estuviera a una distancia euclidiana del punto inicial menos que un valor predeterminado  $\epsilon$  del punto inicial.

En todas las simulaciones de este tipo con  $\epsilon=0.01$  siempre el proceso terminaba comprobándose que las orbitas vuelven al inicio o sea las orbitas periódicas son densas en el torus .

Por ejemplo

**Simulación 1**

Punto inicial de la orbita

[0.38496812159771354, 0.4974119072276715]

Punto de la orbita a distancia menor del punto inicial que 0.01

[0.3812877471920979, 0.5005469063299843]

distancia

2.3373375137e-05

No iteraciones

4226

**Simulación 2**

Punto inicial de la orbita

[0.8913310831730439, 0.7447646354934394]

Punto de la orbita a distancia menor del punto inicial que 0.01

[0.8829110481523208, 0.7489655865381084]

distancia

8.85449794299e-05

No iteraciones

941

**Simulación 3**

Para  $\epsilon=0.001$

Punto inicial de la orbita

[0.003821021084488452, 0.05897761861381068]

Punto de la orbita a distancia menor del punto inicial que 0.001

[0.003722583756684106, 0.05889067874441967]

distancia

1.7248448395e-08

No iteraciones

10483

### **Código en Python**

```
import math
import random
def ArnoldCat(x,y):
    return [math.fmod(x+y,1),math.fmod(x+2*y,1)]
x0=random.random()
y0=random.random()
##print([x0,y0])
[x,y]=ArnoldCat(x0,y0)
eps=0.001
cont=1
while (math.pow((x-x0),2)+math.pow((y-y0),2) > math.pow(eps,2)):
    [x,y]=ArnoldCat(x,y)
    cont=cont+1

print("Punto inicial de la orbita")
print([x0,y0])
print("Punto de la orbita a distancia menor del punto inicial que " +str(eps) )
print([x,y])
print("distancia " )
print(math.pow((x-x0),2)+math.pow((y-y0),2))
print("No iteraciones " )
print(cont)
```