**Curva del dragón.**

Trabajo phyton

*Celia López-Aguado Gálvez*

*Manuel Cendan Allen*

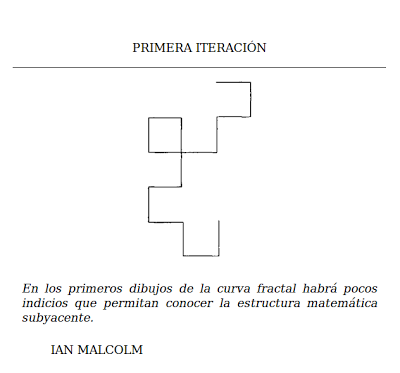
Índice

1. ¿Qué es la curva del dragón?
2. Explicación del código
3. Códigos
4. Fotos
5. Conclusiones del trabajo
6. ¿Qué es la curva del dragón?

La curva del dragón es un fractal que puede construirse así:

* A partir de un segmento, se construye el triángulo rectángulo e isósceles, como lo muestra las dos primeras figuras. Luego se borra el segmento inicial.
* Se repite un sinfín de veces el proceso de remplazar un segmento por otros dos para cada línea de la curva, alternando siempre la orientación de los triángulos.

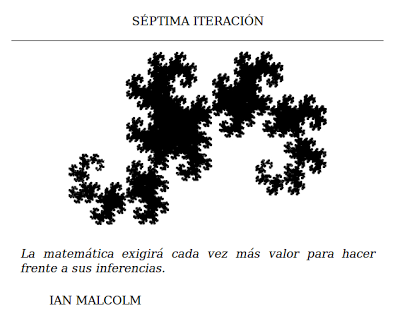
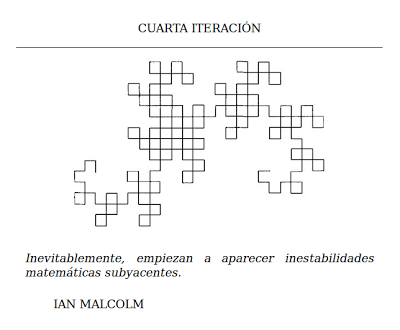
*La curva del dragón de parque jurásico*

El grueso de la novela se divide en siete secciones que el autor llama “iteraciones”. Justo al inicio de cada una de estas iteraciones viene un dibujo junto con una cita atribuida al Dr. Ian Malcolm, uno de los personaje centrales de la novela. Por ejemplo: 

Ian Malcolm es un matemático, experto en teoría del caos. Durante el transcurso de la historia, Malcolm hace varias predicciones basadas en esta teoría sobre las consecuencias negativas que podrían ocurrir al momento de intentar controlar la naturaleza. Lamentablemente, dichas predicciones resultan ser acertadas.

La figura que se muestra al inicio de cada iteración es un fractal conocido como “curva de dragón” o “dragón de Heighway”. Gracias a su aparición en la novela de Crichton, también se le conoce hoy en día como “fractal de Parque Jurásico”. John Heighway, Bruce Banks y William Harter, físicos de la NASA, fueron los primeros en investigar sobre este fractal a mediados de los años sesenta.

En la novela, Malcolm utiliza las curvas de dragón como una manera de simular las acciones “caóticas” que sucederían dentro del parque. Las siguientes dos imágenes muestran las curvas de dragón de la cuarta y séptima iteración, respectivamente. En la segunda figura podemos ver que el fractal tiene una apariencia similar a lo que podría ser un dragón. De ahí es donde toma su nombre.



1. Explicación del código

Podemos usar gráficas de tortuga para dibujar la curva del dragón. Hay varias formas de realizar la implementación, pero una manera sencilla de hacerlo es calcular primero todos los giros que tiene que hacer la tortuga antes de comenzar a dibujar.

El algoritmo consiste en producir una cadena de letras I y D, donde I representa un giro de 90° a la izquierda y D un giro de 90° a la derecha. El pseudocódigo del algoritmo recursivo quedaría así:

Para una curva de orden 0, el resultado es una cadena vacía, ya que no se requiere hacer giro alguno.

Para una curva de cualquier otro orden n, donde n > 0:

Sea c la cadena de la curva de dragón de orden n – 1.

Sea r la cadena en reversa de c.

Sea i la cadena formada por los elementos de r, pero invirtiendo cada I por D y cada D por I.

El resultado consiste en concatenar los siguientes tres elementos: c, I, i.

La traducción directa a Python 3.x se muestra a continuación. Por simplicidad se usan listas de caracteres para representar las diversas cadenas que aparecen en el pseudocódigo en lugar de utilizar strings.

def curva(n):

"""Devuelve una lista de caracteres con los giros

requeridos para dibujar la curva de dragón de orden

n. La letra 'I' significa girar a la izquierda,

mientras que la letra 'D' significa girar a la

derecha.

"""

if n == 0:

return []

else:

c = curva(n - 1)

r = c[::-1]

i = ['I' if g == 'D' else 'D' for g in r]

return c + ['I'] + i

El siguiente código usa la función curva() para mostrar los giros requeridos con el fin de dibujar las curvas de dragón de órdenes 0 al 5:

for i in range(6):

print('orden {0}: {1}'.format(i, ''.join(curva(i))))

La salida sería:

orden 0:

orden 1: I

orden 2: IID

orden 3: IIDIIDD

orden 4: IIDIIDDIIIDDIDD

orden 5: IIDIIDDIIIDDIDDIIIDIIDDDIIDDIDD

Una vez que tenemos los giros necesarios para la curva de dragón resulta muy directo el mecanismo para realizar el dibujo usando gráficas de tortuga:

Mover la tortuga hacia adelante x pixeles.

Por cada elemento g en la lista de giros de la curva de dragón de orden n, hacer lo siguiente:

Si g es I, girar la tortuga 90° a la izquierda.

Si g es D, girar la tortuga 90° a la derecha.

Mover la tortuga hacia adelante x pixeles.

El código correspondiente en Python queda así:

from turtle import \*

def dragon(n, x):

"""Dibuja una curva de dragón de orden n en donde

cada segmento de la curva es de longitud x.

"""

fd(x)

for g in curva(n):

if g == 'I':

lt(90)

else: # g == 'D'

rt(90)

fd(x)

Agregando un poco de código para inicializar la tortuga de manera conveniente, ya tenemos un programa que dibuja una curva de dragón de orden 8:

hideturtle()

pensize(3)

color('SteelBlue')

speed('fastest')

setheading(180)

dragon(8, 15)

done()

1. Código completo

def curva(n):

if n == 0:

return []

else:

c = curva(n - 1)

r = c[::-1]

i = ['I' if g == 'D' else 'D' for g in r]

return c + ['I'] + i

for i in range(6):

print('orden {0}: {1}'.format(i, ''.join(curva(i))))

from turtle import \*

def dragon(n, x):

fd(x)

for g in curva(n):

if g == 'I':

lt(90)

else: # g == 'D'

rt(90)

fd(x)

hideturtle()

pensize(3)

color('SteelBlue')

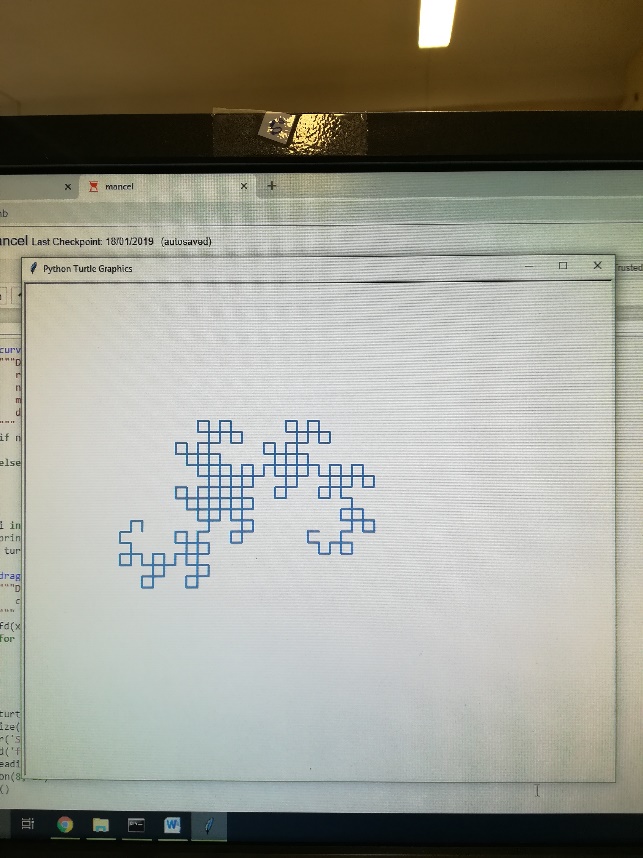
speed('fastest')

setheading(180)

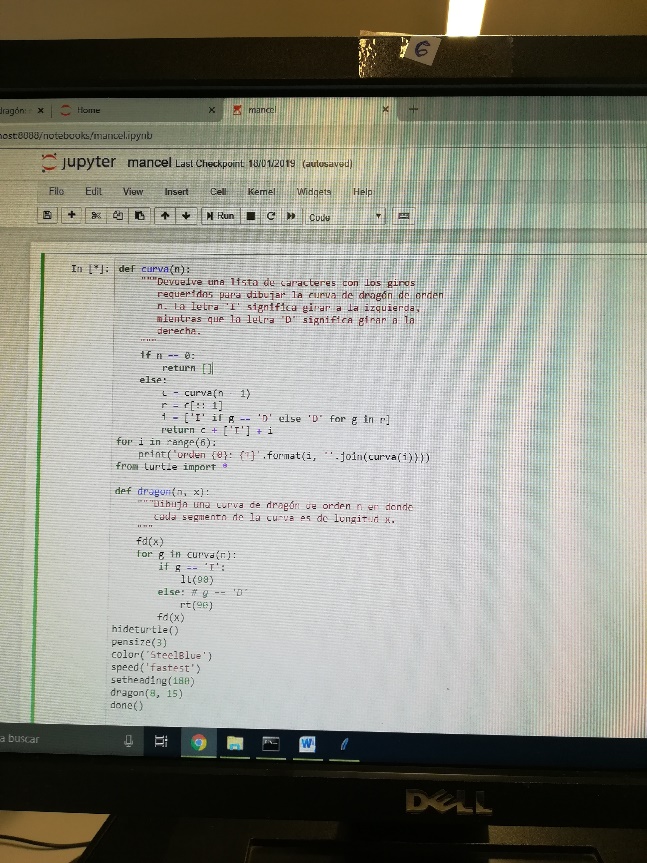
dragon(8, 15)

done()

1. Fotos



curva del drogón en Python con turtle



1. Conclusiones

Solo fue posible representar con Python la curva del dragón “original” puesto a que al intentar modificar este código daba error, nombrar también que para trabajar con fractales como es la curva del dragón hay que tener más conocimientos sobre fractales y hacer una investigación más extensa y exhaustiva. Quitando eso fue un trabajo interesante de hacer y de leer por toda la información relativa ha dicho código, además de la explicación que dan sobre la relación del código relacionado con Parque Jurásico de Michael Crichton y publicada en 1990.