

# Proyecto Python

## Fractales

Fernando García Ballesteros  
Saúl Lozano Selva

---

# Proyecto Python: Fractales

---

## Índice

INTRODUCCIÓN .....	2
CÓDIGO .....	2
EXPLICACIÓN DEL CÓDIGO Y RESULTADO .....	3
CONCLUSIÓN .....	4

## INTRODUCCIÓN

En esta ocasión hemos realizado un proyecto de programación con la aplicación de python, dando como resultado un fractal de colores y forma aleatoria partiendo siempre del mismo origen y creando en todas la ocasiones una especie de explosión que recuerda a un fuego artificial.

## CÓDIGO

```
import turtle
import random
turtle.tracer(0, 0)
wn = turtle.Screen()
wn.colormode(255)
turtle.bgcolor("black")
alex = turtle.Turtle()
alex.speed(10)
alex.goto(0,0)
alex.pensize(0)
alex.ht()
for i in range(400):
    alex.color(random.randrange(256),random.randrange(256),random.randrange(256))
    alex.goto(round(random.gauss(0,100),0),round(random.gauss(0,100),0))
    x = alex.xcor()
    y = alex.ycor()
    for j in range(25):
        s = round(random.gauss(0,5), 0)
        t = round(random.gauss(0,5), 0)
        alex.color(random.randrange(256),random.randrange(256),random.randrange(256))
        alex.pensize(0)
        alex.goto(x + s, y + t)
    alex.goto(s,t)
turtle.update()
wn.exitonclick()
```

## EXPLICACIÓN DEL CÓDIGO Y RESULTADO

Hemos dividido el código en 3 partes para facilitar así su explicación.

```
import turtle
import random
turtle.tracer(0, 0)
wn = turtle.Screen()
wn.colormode(255)
turtle.bgcolor("black")
alex = turtle.Turtle()
alex.speed(10)
alex.goto(0,0)
alex.pensize(0)
```

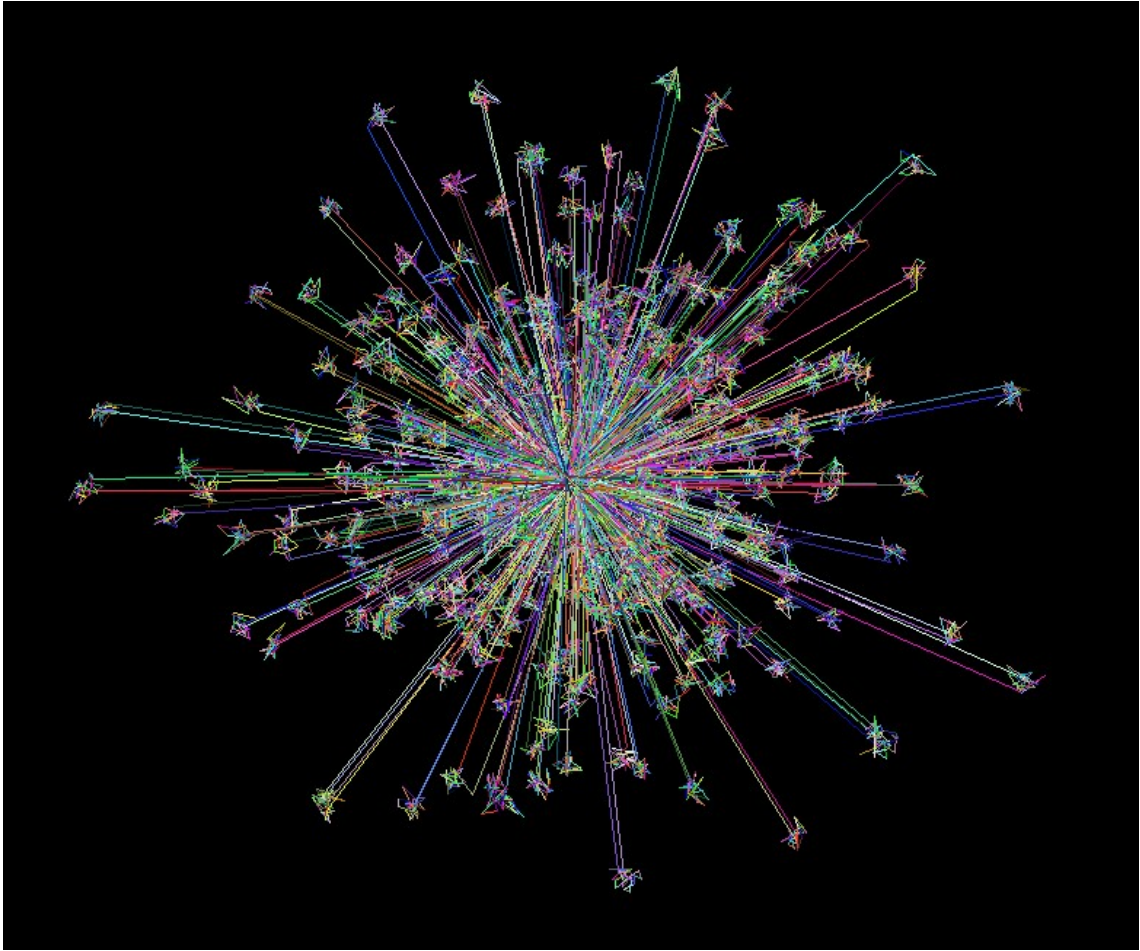
En esta parte del código, importamos la librería denominada como turtle (apreciable en la primera instrucción) y otra librería aleatoria (apreciable en la segunda instrucción). Posteriormente en las dos siguientes filas indicamos que aparezca la pantalla de dibujo y en la posición que queremos que lo haga. En la sexta línea hacemos que el fondo de la pantalla de dibujo sea negro para que así se vea mejor el fractal. Para poder dibujarlo en las últimas cuatro líneas nombramos a la figura como Alex facilitando su programación y configuramos la velocidad de dibujo, el punto en el que ha

```
for i in range(400):
    alex.color(random.randrange(256),random.randrange(256),random.randrange(256))
    alex.goto(round(random.gauss(0,100),0),round(random.gauss(0,100),0))
    x = alex.xcor()
    y = alex.ycor()
```

En este conjunto de instrucciones, para el parámetro i, en un rango de 400 unidades, establecemos un color aleatorio en el 256 y una posición de los fractales que sea aleatoria de 0 a 100. De esta forma y definiendo las constantes x e y dentro de la variable Alex, los fractales saldrán de forma aleatoria en color y posición cada vez que ejecutemos el código.

```
for j in range(25):
    s = round(random.gauss(0,5), 0)
    t = round(random.gauss(0,5), 0)
    alex.color(random.randrange(256),random.randrange(256),random.randrange(256))
    alex.pensize(0)
    alex.goto(x + s, y + t)
alex.goto(s,t)
turtle.update()
wn.exitonclick()
```

En este conjunto, para el parámetro j, en un rango de 25 unidades, establecemos que para cuando la línea termine su recorrido dibuje un pequeño “garabato” de colores aleatorios. Por ultimo actualizamos la ventana de dibujo y finalizamos el programa.



## CONCLUSIÓN

Como conclusión, decir que se obtiene una forma irregular, totalmente aleatoria, pero formando en todos los casos una forma bastante similar a un fuego artificial.

Para este proyecto nos hemos basado de internet modificándolo en numerosas ocasiones y probando tantas otras debido a nuestros limitados conocimientos con este programa, ya que se trata de nuestro primer trabajo con este, dando como resultado esta especie de explosión multicolor.