

# Trabajo Práctico: Conjunto de Julia

Abarello, Guido, *Padrón Nro. 97.119*  
guidosergio12@gmail.com

Kim, Diego, *Padrón Nro. 94.783*  
diegofk26@gmail.com

Prystupiuk Maximiliano, *Padrón Nro. 94.853*  
mprystupiuk@gmail.com

2do. Cuatrimestre de 2016  
66.20 Organización de Computadoras – Práctica Martes  
Facultad de Ingeniería, Universidad de Buenos Aires

## Abstract

En este artículo se expone la implementación de un programa en C con el objetivo de graficar el conjunto de Julia y sus vecindades, exportándolo en formato PGM al archivo de salida, en el caso de indicarlo.

## 1 Introducción

Los conjuntos de Julia son una familia de conjuntos fractales que se obtienen al estudiar el comportamiento de los números complejos al ser iterados por una función holomorfa.

Una familia muy notable de conjuntos de Julia se obtienen a partir de funciones cuadráticas simples  $f_c(z) = z^2 + c$ , donde  $c$  es un número complejo.

Para todo complejo  $z$ , se construye la siguiente sucesión:

$$\begin{aligned} z_0 &= z \\ z_{n+1} &= z_n^2 + c \end{aligned}$$

Si esta sucesión queda acotada, entonces se dice que  $z$  pertenece al conjunto de Julia de parámetro  $c$ , caso contrario queda excluido.

El programa desarrollado en este trabajo práctico calcula la sucesión para cada punto de una región del plano complejo especificada por parámetro, y le asigna a ese punto un valor en escala de grises según la cantidad de iteraciones necesarias hasta que se dé una condición de corte especificada sobre el valor de  $z$ . Finalmente, guarda éstos datos en formato de imagen PGM.

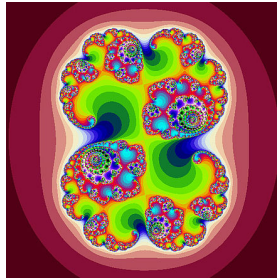


Figure 1: Conjunto de Julia.  $C = [0.285, -0.01]$ .

## 2 Diseño e implementación

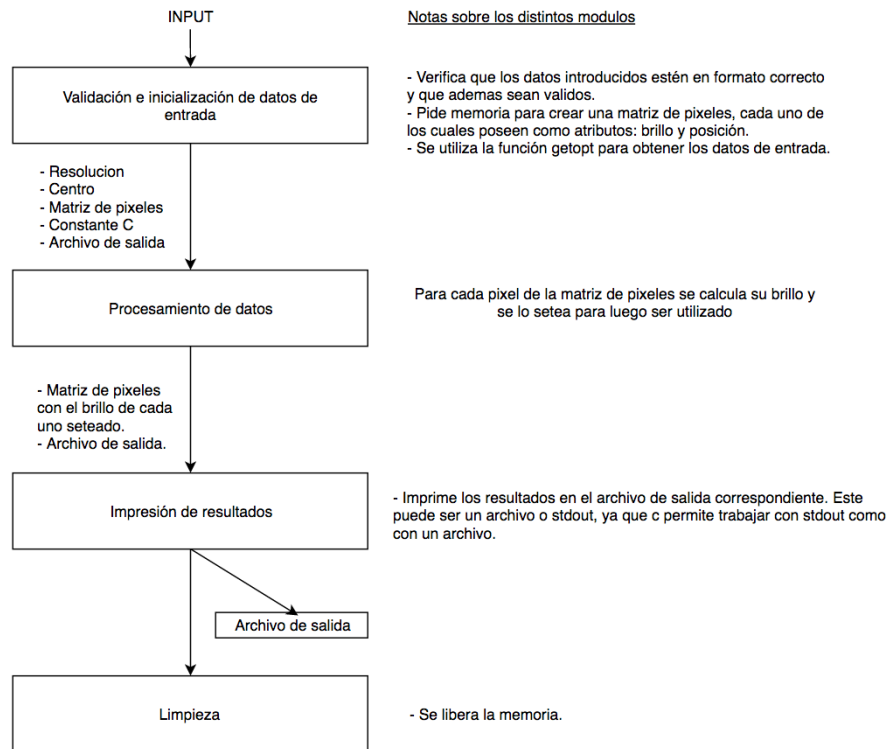
Se calculan los valores en base a los parámetros dados por línea de comandos, y especificados en el enunciado del trabajo práctico, el punto del plano complejo que se asigna a cada píxel. Se procede a realizar los cálculos mencionados para todos los píxeles considerados.

Se tomaron las siguientes consideraciones:

- Si en alguna iteración  $|z^n| > 4$  se sabe que la sucesión diverge.
- Si la cantidad de iteraciones es igual a 255 se considera que el punto pertenece al conjunto de Julia.

Se guarda la cantidad de iteraciones obtenida para cada píxel de la imagen de acuerdo al formato PGM, para obtener una representación gráfica del conjunto. El código fuente necesario para correr la aplicación se encuentra en el anexo A.

A continuación se mostrará un diagrama que intenta explicar los distintos módulos que intervienen en el programa:



### 3 Compilación

Se proveerá la carpeta 'src', que se deberá copiar a NetBSD de la manera, y bajo las condiciones especificadas por la cátedra.

Para compilar el programa se dispone de un archivo de tipo makefile el cual se deberá ejecutar la siguiente manera

```
#guestOS gmake
```

#### 3.1 Ejecución de pruebas

Se provee, además un shell script que realiza pruebas automáticas con distintos parámetros de entrada. Para ejecutar las pruebas, luego de compilar el programa se debe ejecutar:

```
#guestOS ./runTests.sh
#guestOS ./runIntegrationTests.sh
```

Dichas pruebas son autodescriptivas y detallan en tiempo de ejecución tanto el input como el resultado obtenido para cada una de las pruebas.

#### 3.2 Comando de ayuda

Ejecutando el programa de la siguiente manera:

```
#guestOS ./ejecutable -h
```

Se mostrará una lista de los comandos disponibles y su descripción.

## 4 Conclusiones

Una de las conclusiones más importantes fue la notoria diferencia de performance entre el GuestOS y el HostOS. Siendo el guestOS mucho más lento, debido a su emulación. Por este motivo creemos que una mejora en el programa sería mucho más notoria en este último.

## References

- [1] Conjunto de Julia, <https://es.wikipedia.org/wiki/ConjuntoDeJulia>
- [2] J. L. Hennessy and D. A. Patterson, “Computer Architecture. A Quantitative Approach,” 3ra Edición, Morgan Kaufmann Publishers, 2000.
- [3] J. Larus and T. Ball, “Rewriting Executable Files to Measure Program Behavior,” Tech. Report 1083, Univ. of Wisconsin, 1992.