

**Aluno:** Gustavo Henrique Roos e Franco Noronha Pereira

**Matrículas:** 20101184 e 20100524

**Data:** Outubro de 2022

### **Implementação com visualização gráfica e duas linguagens de programação**

A aplicação escolhida e desenvolvida foi o conjunto de Mandelbrot. O conjunto tem o nome de Benoit Mandelbrot, que foi o primeiro a estudá-lo e possui muitas propriedades interessantes, incluindo auto-semelhança e uma dimensão fractal. É gerado pela recursão da função complexa:

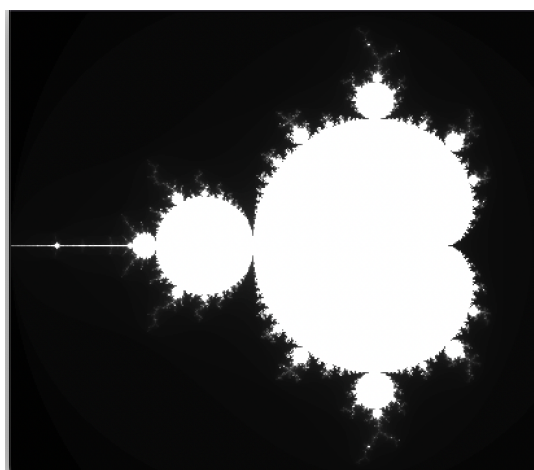
$$z_0 = 0$$

$$z_{n+1} = z_n^2 + c$$

*não* tende ao infinito.

onde  $z$  é um número complexo e  $c$  é uma constante complexa. A função é iterada começando com  $z = 0$ . O conjunto de Mandelbrot (Figura 1) é o conjunto de todos os números complexos  $c$  para os quais a iteração não diverge.

Figura 1: Ilustração do conjunto de Mandelbrot



Fonte: própria autoria (2022)

As linguagens escolhidas para o desenvolvimento foram *Python* como linguagem de alto nível e *C++* para os cálculos das operações complexas. A integração entre as linguagens foi realizada através de uma biblioteca dinâmica criada em *C++*

utilizando a diretiva *-shared* no comando de compilação, gerando um arquivo com a extensão *.dll* (no *Windows*) ou *.so* (no *MacOS* e *Linux*), que basicamente é um arquivo binário executável (*.exe*) que pode ser invocado por algum componente – no caso o programa escrito em *Python* –, a qualquer momento. A biblioteca gerada pode ser acessada no *Python* através de sua biblioteca nativa *ctypes*, que torna as funções definidas na biblioteca acessíveis no código e lida com os problemas de compatibilidade de tipos entre as linguagens automaticamente. Nas funções da biblioteca criamos e alteramos os valores de um array que é compartilhado entre as linguagens, a fim de realizar os cálculos para cada ponto no plano dos complexos e acessar os valores na interface gráfica em *Python*.