

**UNIVERSIDADE FEDERAL DE PELOTAS  
CENTRO DE DESENVOLVIMENTO TECNOLÓGICO  
BACHARELADO EM CIÊNCIA DA COMPUTAÇÃO  
CONCEITOS DE LINGUAGEM DE PROGRAMAÇÃO  
PROF. GERSON CAVALHEIRO**



**DOCUMENTAÇÃO DA IMPLEMENTAÇÃO DA APLICAÇÃO GRÁFICA  
UTILIZANDO DUAS LINGUAGENS DE PROGRAMAÇÃO**

LUCAS SIMÕES FERREIRA - 20102119 - lsferreira@inf.ufpel.edu.br  
JUATHAN COELHO DUARTE - 19200989 - juathan.cduarte@inf.ufpel.edu.br

## 1. Especificação

A proposta do trabalho é implementar uma aplicação gráfica tendo como sugestão, o Fractal de Mandelbrot, Ray Tracing ou Simulação de Partículas, com apoio de duas linguagens de programação: Python e C (ou C++). O desafio consiste em realizar o uso conjunto de duas linguagens de programação, sendo o desafio especificamente, realizar o uso conjunto das linguagens. Python deve ser utilizado para oferecer uma interface com o usuário e apresentar a imagem gerada. A linguagem C (ou C++) deve ser utilizada para implementar o serviço de cálculo desejado.

## 2. Aplicação

Escolhemos implementar o **Fractal de Mandelbrot**, fazendo os cálculos e gerando o Fractal na linguagem C em paralelo com Python, gerando nossa interface gráfica.

Na matemática, **conjunto de Mandelbrot** é um fractal definido como o conjunto de pontos  $c$  no plano complexo para sequência é definida recursivamente, tal sequência se expande como:

$$\begin{aligned}c &= x + iy \\Z_0 &= 0 \\Z_1 &= Z_0^2 + c \\&= x + iy \\Z_2 &= Z_1^2 + c \\&= (x + iy)^2 + x + iy \\&= x^2 + 2ixy - y^2 + x + iy \\&= x^2 - y^2 + x + (2xy + y)i \\Z_3 &= Z_2^2 + c = \dots\end{aligned}$$

Em sua representação gráfica, pode ser dividido em um conjunto infinito de figuras, sendo a maior delas um cardióide (curva que pode ser produzida como um locus) localizado ao centro do plano complexo. Existe uma infinidade de quase-círculos que tangenciam o cardióide e variam de tamanho com raio tendendo assintoticamente a zero.

Quando se explora o **Conjunto de Mandelbrot** com mais resolução encontram-se sempre réplicas e mais réplicas do conjunto, sendo uma característica dos objectos fractais.

Para a interface utilizamos da ferramenta **QtDesigner**, onde geramos os arquivos .ui (XML), logo após utilizamos da biblioteca chamada PyQt5 para converter o .ui para .py e começamos a desenvolver o restante da lógica.

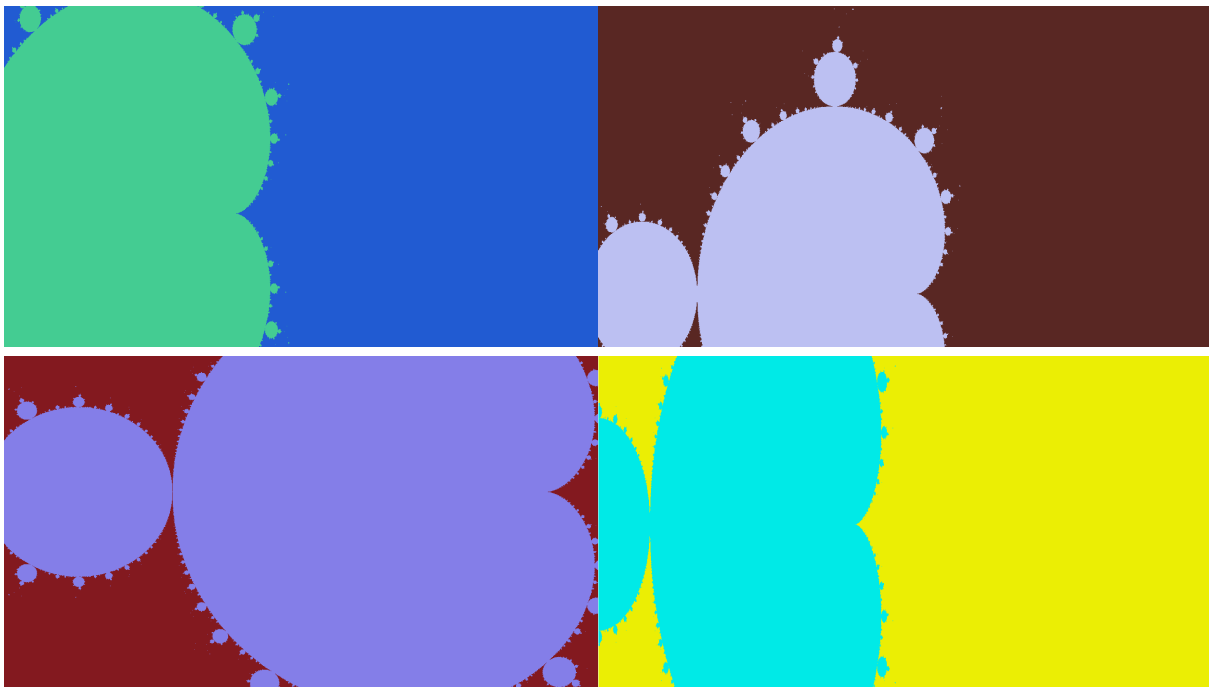
Já no programa na linguagem C, geramos o fractal desejado, possuindo apenas a função *main()*, na qual calculamos valores aleatórios usados para que as imagens não se repitam durante o uso da aplicação, tanto em posição quanto em cor do Fractal. Devemos também possuir um arquivo .h deste código, declarando todas as funções, neste caso, possui apenas a declaração da *main()* já que é nossa única função. Com isso geramos a biblioteca compartilhada, pelo comando “gcc -shared -o libMandelbrot.so -fPIC mandelbrot.c”, criando o arquivo *libMandelbrot.so*.

Este arquivo é um “*shared object*”, são bibliotecas ligadas dinamicamente, onde os módulos usados pelo programa são carregados para memória em tempo de carregamento ou execução ao invés de serem copiados pelo ligador quando criado seu executável.

Assim no Python, no código do arquivo *fractalMandelbrot.py*, utilizamos da biblioteca “ctypes” com o comando “*CDLL(nome\_da\_biblioteca\_compartilhada.so)*” atribuindo-o para uma variável, assim podendo utilizar as funções da biblioteca C.

No código há comentários explicando as funções e trechos principais do código para melhor entendimento do funcionamento.

### 3. Alguns Fractais Gerados



### 4. Revisão Bibliográfica

- 1.Mandelbrot Set ([https://pt.wikipedia.org/wiki/Conjunto\\_de\\_Mandelbrot](https://pt.wikipedia.org/wiki/Conjunto_de_Mandelbrot))
- 2.Mandelbrot Set in all in all programming languages ([https://rosettacode.org/wiki/Mandelbrot\\_set#C](https://rosettacode.org/wiki/Mandelbrot_set#C))
- 3.Fractal in C/C++ (<https://www.geeksforgeeks.org/fractals-in-cc/>)
- 4.Calling C Functions from Python [Step by Step Procedure] (<https://www.csestack.org/calling-c-functions-from-python/>)

## 5. Suporte

1. Pyinstaller adding data files  
(<https://stackoverflow.com/questions/41870727/pyinstaller-adding-data-files>)
2. Pyinstaller manual (<https://pyinstaller.org/en/stable/index.html>)
3. Introduction to makefile  
(<https://embarcados.com.br/introducao-ao-makefile/>)