

Geração de fractal de Mandelbrot com interface

IMPORTANTE: TODO O TRABALHO FOI CONSTRUÍDO UTILIZANDO WINDOWS.

Aplicação:

A aplicação consiste em uma interface, em python, que possibilita a geração de uma imagem ppm de uma fractal de Mandelbrot que utiliza uma biblioteca compartilhada implementada em C (*o código da fractal de Mandelbrot é de autoria de Andrej Bauer, disponível em gist.github.com/andrejbauer/7919569*).

Implementação:

O trabalho desenvolveu a interface utilizando a biblioteca **PySimpleGUI**, que possibilitou a criação do layout, que consiste nos campos de entrada de dados, botão para entrada dos parâmetros e a exibição da imagem da fractal gerada.

Além disso o código de interface também possui a biblioteca **ctypes**, que viabiliza a compatibilidade com dados da linguagem C, e a chamada de funções através de DLLs ou bibliotecas compartilhadas.

O código em C, que é a fractal de Mandelbrot, sofreu poucas alterações, as alterações apenas se referem a renomeação da classe e adição dos parâmetros. O código gera uma imagem .ppm e salva na pasta raíz.

A compilação desse código em C recebe argumentos que geram um arquivo .so, ele resulta em uma biblioteca compartilhada, onde será posteriormente acessada pelo código em python utilizando “ctypes.CDLL()”, que é a função utilizada para fazer o carregamento das bibliotecas. Com isso é possível acessar todos os métodos presentes na biblioteca, no caso desse trabalho, apenas uma função.

Para finalizar a comunicação, os campos preenchidos na interface em python são obtidos através do “PySimpleGUI.Window(layout).values.values[campo]”, e passados utilizando o objeto retornado no parágrafo anterior “ctypes.CDLL(mlibExt.so)”, com a função “argtypes” onde são especificados os tipos de dados, e, depois, a própria função da biblioteca e, finalmente, os dados. E, então, é exibida a imagem .ppm gerada na interface.

Métodos:

mandelbrotExt.c:

mandelbrotExtExecute – Método que recebe a entrada dos valores para a fractal, e gera uma imagem .ppm (que é a fractal de mandelbrot) na pasta raíz.

interface.py:

main() - Método que cria o objeto windows (interface), cria o objeto da biblioteca compartilhada, e possui o loop de execução do programa.

Comunicação entre as linguagens:

A biblioteca Ctypes, do Python, fora utilizada para ter o acesso as funções criadas no código em C. Para isso, anteriormente foi preciso compilar o código em C utilizando as flags “-shared” e “-fPIC”, que indica a criação de uma biblioteca compartilhada, e que essa biblioteca não necessita estar em um endereço virtual específico, respectivamente.

Dentro do código em python foi utilizado o método ctypes.CDLL que, uma instância desse objeto, é uma biblioteca carregada.

E o método ctypes.(chamadaDoMétodo).argtypes, que deve ser chamado antes de ser executada a função, pois necessita especificar os tipos dos argumentos. Quando a função é chamada os itens na tupla argtypes possibilitam que seja feita a conversão de tipos. No caso dos tipos do python, para os tipos em C utilizando as regras de conversão.