CS293 Project : Mandelbrot Zoom

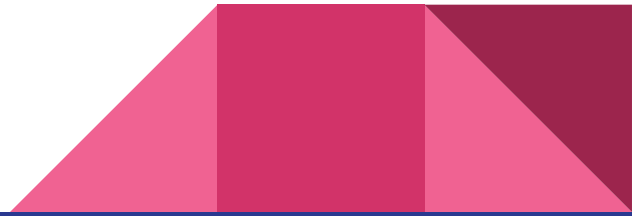
By Adish Shah
Roll number : 200020012

Overview

1. Interactive Mandelbrot Zoom implemented using SFML graphics lib
2. Mandelbrot animation video
3. Cycle Detection Algorithms

All the files required are in the link :

https://drive.google.com/drive/folders/11TNLOiSgG6gwRmmexCYb4dCwqKvntYxw?usp=sharing



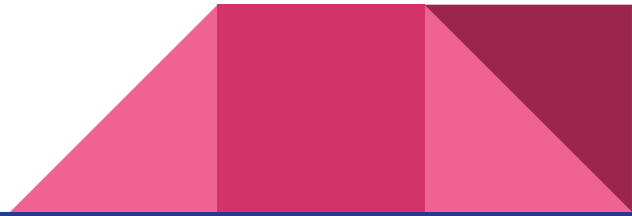
Interactive Mandelbrot Zoom implemented using SFML graphics lib

1) Added keyboard and mouse controls to zoom in, zoom out, move left, right, up and down, and increased, decrease iterations.
2) Added feature to **reset to the beginning** and **undo** last move.
3) Used **Queue Data Structure** to store the keyboard keys when pressed and added a feature to play from start, which will replicate all the moves of user from the last reset.

   How to run :

   $ make -f part1_make

   $ ./interactive_mandelbrot



DEMO

Mandelbrot Animation Video

1) Implemented Mandelbrot set in 2 ways : one using normal **long double** .
2) Secondly using, custom **double double** math to **increase precision**.
3) Can generate a high resolution image with even higher zoom
4) Wrote a script to generate a **video** of Mandelbrot set by continuously zooming in at chosen point.

Instructions to run :

$ make

$ bash run.sh 1.0 1.0e+30 14400 mandel_dd 1920 1080 -0.74364388703715870475219150611477 4 0.13182590420531197049313205638513 9

$ cd build_video/

$ ffmpeg -i out_%05d.png out.mp4

DEMO

Cycle Detection Algorithms

1) Implemented Brent’s algorithm, Floyd’s tortoise and hare algorithm.

Cycle detection or cycle finding is the algorithmic problem of finding a cycle in a sequence of iterated function values.

This has application in periodicity checking in Mandelbrot set for optimising the time taken to plot the mandelbrot image.

"Periodicity Checking" technique consists on detecting the cycles when we are calculating the orbit for a point c, that way we can stop iterations, because we know that the series is not going to diverge.

https://en.wikipedia.org/wiki/Cycle_detection

Requirements :

Number of lines of code written : Total = 1255

mandelbrot_interactive.cpp	550
DynamicDeque.cpp	141
doubledouble.h	164
mandel.cpp	102
mandel_dd.cpp	140
brents_alg.cpp	39

floyds_alg.cpp	51
run.sh	48
make_image.sh	5
Makefile	7
part1_make	8

Thank you