Lab Report

Mahmoud Natsheh

1001860023

CSE 3320 – 003

Dr.Bakker

Spring 2022

Purpose of the experiments and the experimental setup:

The purpose of this experiment is to demonstrate how splitting up the task on to multiple CPUs or threads can help speed up the time taken by the computer to process a specific task. The experimental setup consist of multithreading the program given in order to create multiple threads that will run the program in order to speed the process speed. First I ran the program with the command line parameters of: ./mandel –x 0.286932 –y 0.014287 –s .000001 –H 5000 –m 2000 –o mandel2.bmp

This command line basically tells the program that the x coordinate is 0.286932, the y coordinate is 0.014287 and scale the image by 0.000001, the height of the image is 5000, the magnification is 2000, run the image with 1 thread, then save the resultant image as mandel2.bmp. This resulted in the program executing in 5.14 seconds.

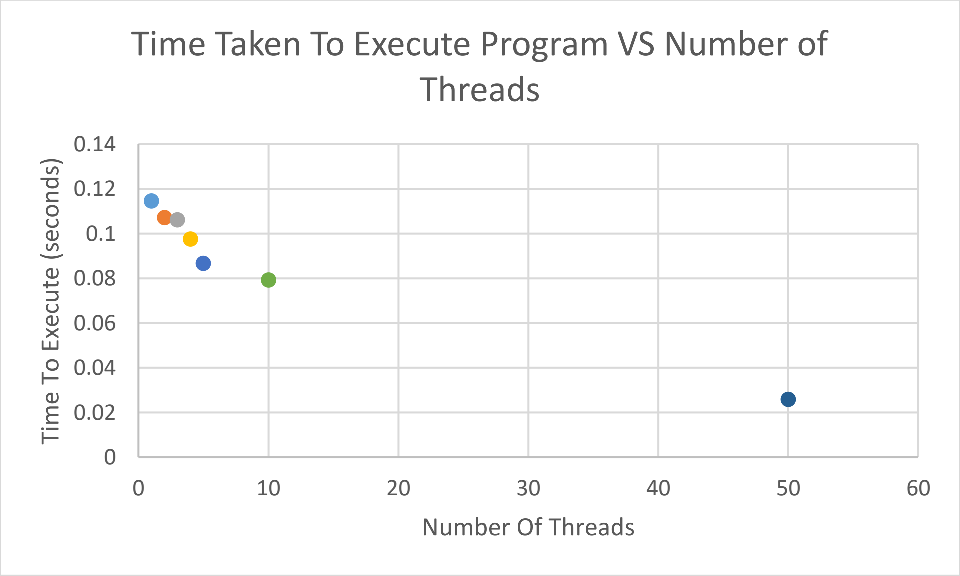
Next, I tried the same image, however I ran it with 100 thread using the command line: ./mandel –x 0.286932 –y 0.014287 –s .000001 –H 5000 –m 2000 –o mandel2.bmp –n 100

This resulted in the program executing in 0.49 seconds.

Therefore, this goes to show that a program is able to execute more efficient and quick if it is run with multiple threads.

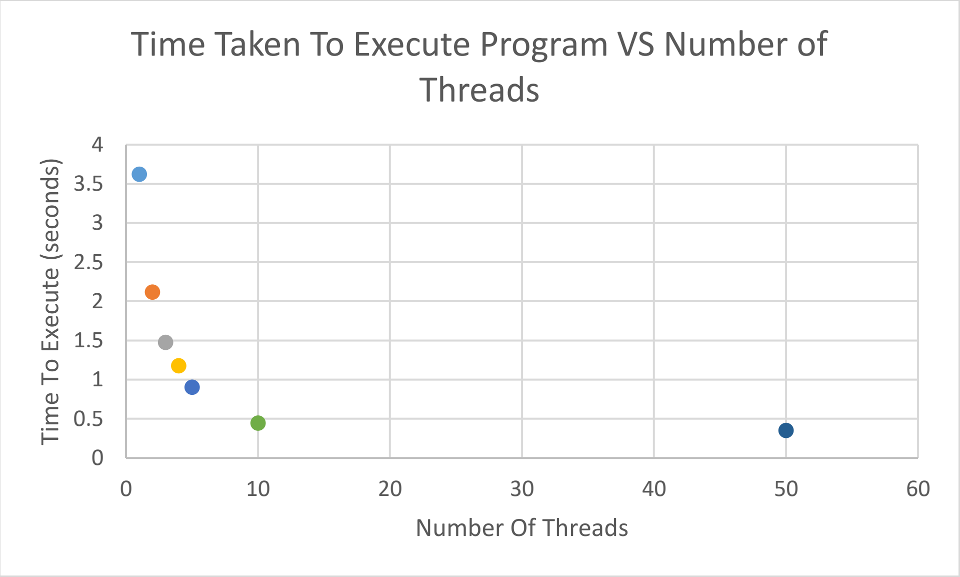
For the command line parameter: ./mandel -x -.5 -y .5 -s 1 -m 2000 the following chart (Configuration A) demonstrates how fast it took to execute the program with 1, 2, 3, 4, 5, 10, and 50 threads.

Configuration A:



For the command line parameter: ./mandel -x 0.2869325 -y 0.0142905 -s .000001 -W 1024 -H 1024 -m 1000 the following chart (Configuration B) demonstrates how fast it took to execute the program with 1, 2, 3, 4, 5, 10, and 50 threads.

Configuration B:



Explain the shape of the two curves, what is the optimal number of threads, why do curves A and B have a different shape?

Curve A (Configuration A) has an almost linear shape moving to the bottom right of the graph and then one point that is all the way at the bottom right of the graph. This shows that each thread that is added to execute the program slowly helps speed up the execution of the program.

Curve B (Configuration B) has an exponential looking shape moving to the bottom right of the graph. This shows that each thread that is added to execute the program instantly helps speed up the execution of the program to the point that it will half the time taken to execute the program.

It seems that the optimal number of threads for Configuration A is one thread. The reason for this is because the graph shows that the more threads added continuously makes the execution time get smaller, but not too small making the optimal be the first point since it is optimal itself. However, for Configuration B it is 10 threads. This is because when 10 threads are added it makes almost the same amount of execution time then 50 threads.

Curve A and B have different shapes because they are operating on two different parameters of the image. This is seen when the second parameter is enlarging the width and height, causing for the program to execute slower since a bigger task is being done. However, Curve A is executing simple things like just moving the image, thus causing for the execution time to be very small to start off with and the more threads we add won’t really help as it is already executing at the optimal number.