|  |  |
| --- | --- |
| Name: | Edward Eisenberger |
| ID# | 1066164 |
| Assignment 4 | |
| Date of Submission | June 14, 2019 |

Table of Contents

[Overview 3](#_Toc10816345)

[Part 1: ReaderWriterLock 3](#_Toc10816346)

[Summary 3](#_Toc10816347)

[Results 4](#_Toc10816348)

[Part 2: TPLTest 5](#_Toc10816349)

[Summary 5](#_Toc10816350)

[Results 5](#_Toc10816351)

# Overview

Assignment 4 consisted of implementing the in the Parallel Programming Examples using TPL Handout as well as implementing a parallel Swarm Optimization solution using TPL to find all the roots of Himmelblau’s function.

# Part 1: Parallel Programming Examples using TPL Handout

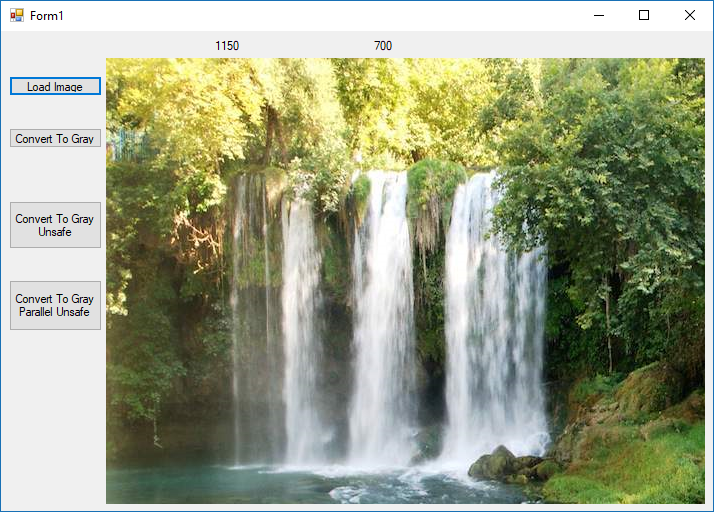
## Summary

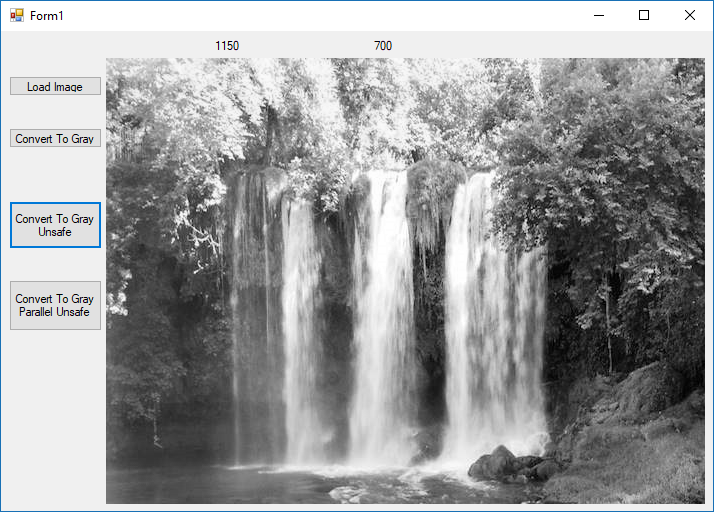
Part 1 consisted of implementing four separate projects: ImageProcessing, MatrixMultiplication, ParallelQuickSort, and ParallelSwarmIntelligence.

## Results

The results of each project are shown below.

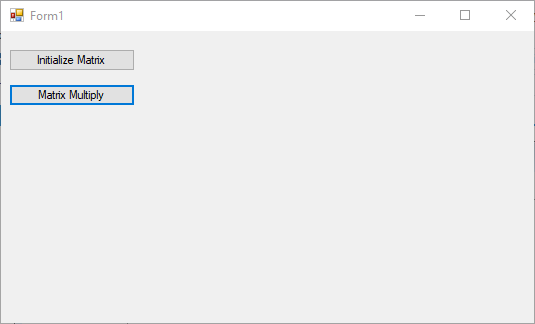
### ImageProcessing

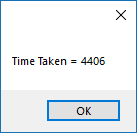




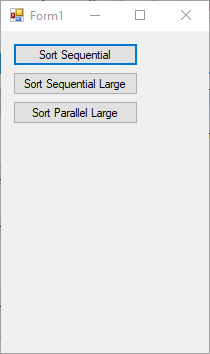
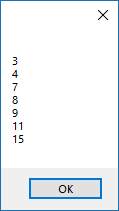
|  |  |  |
| --- | --- | --- |
| Figure 1 - Convert To Gray | Figure 2 - Convert To Gray Unsafe | Figure 3 - Convert To Gray Parallel Unsafe |

### MatrixMultiplication



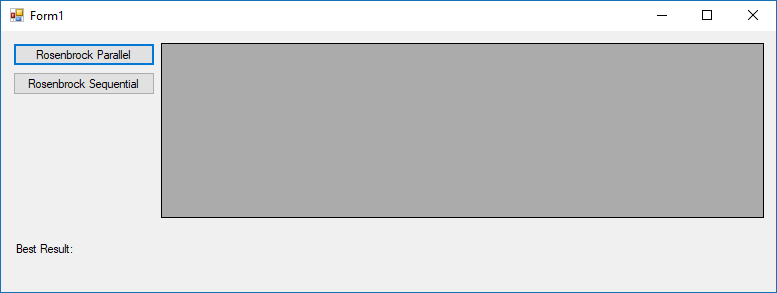


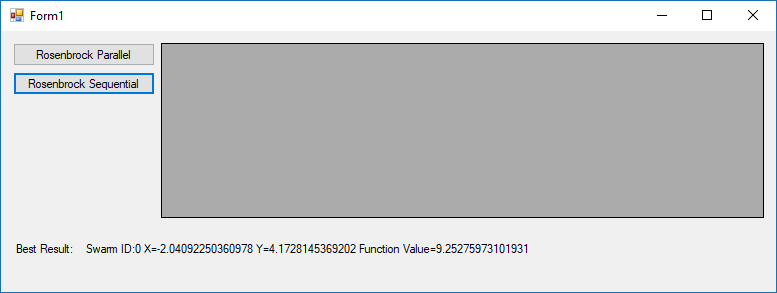
### ParallelQuickSort

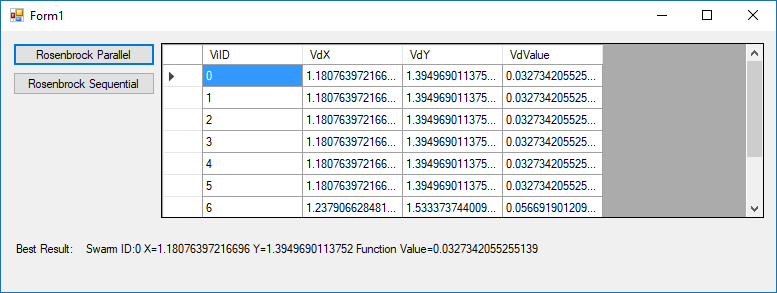
 

|  |  |
| --- | --- |
| Figure - Sort Sequential Large | Figure - Sort Parallel Large |

### ParallelSwarmIntelligence







# Part 2: Parallel Swarm Optimization for Himmelblau’s Function

## Summary

Part 2 consisted developing and implementing a Parallel Swarm Optimization solution using TPL to find all roots of Himmelblau’s function. This was done using the ParallelSwarmIntelligence project as a basis. The project was modified such that the SwarmSystem could be initialized with difference ranges and work on Himmelblau’s function. An additional button, Himmelblau Parallel, was added to the form which executes 80 swarms in parallel. Unique results are stored in list and used as a DataSource for the DataGridView. Results are considered unique if their calculated value is less than 0.5 and the Euclidean Distance between all unique results is greater than 1.0.

Using this approach, all 4 roots are found.

## Results

