Parallel Functional Programming Final Project

Tian Low (ttl2132) Selena Huang (sh3696)

November 2020

1 Introduction

The project we would like to work on is a nonogram solver in Haskell. A nonogram, also known as picross, is a rectangle filled with squares that a user will either shade or not shade in, typically in order to complete a picture. The user is given a set of numbers in each row that state how many squares are filled in and in what order. If there is more than one number in the set, this indicates that there is at least one blank square in between both number of squares.

This will require a two key components: a GUI interface and the puzzle solving algorithm.

2 GUI Interface

We are adding a GUI component increase the sophistication of our project. For the GUI Interface, we plan to use the gtk3 package. The user will first input the number of rows and number of columns into the window, then receive a second page with multiple text areas, with each text area representing a row. The user will then receive a third page with multiple text areas, with each text area representing a column. The final page will contain the solution to the nonogram puzzle. A sample of user input is shown below, with each bullet representing a new page.

- Number of rows: 5
  Number of columns: 5

- Row information (space-separated)

    - 1 3
    - 5
    - 2 2
    - 5
    - 5

- Column information (space-separated)

- – 5
- – 4
- – 2 2
- – 5
- – 5

- Solution (with an x representing a shaded box)

x		x	x	x
x	x	x	x	x
x	x		x	x
x	x	x	x	x
x	x	x	x	x

3   Puzzle Solving

We will check if the puzzle generated is a valid square before attempting to solve the puzzle. For example, if the numbers given by the user exceed the number of squares in a row, or the numbers given are impossible to solve with, the user will receive an error message.

We will use the Par Monad for parallelizing the nonogram solving algorithm. We believe that this project will properly be in scope, because we will need to use backtracking (an application of AI) in order to solve the puzzle.