The University of Hong Kong

COMP3258: Functional Programming

Final Project (Jigsaw Sudoku Game)

Final Report

You should write a short final report (in pdf format) that:

1. Describes how to build your project. It is highly recommended that your project builds with 1 or 2 commands (for example by employing a Makefile or some other build scripts). If your project cannot be easily build, you may be penalized.

2. Describes the functionality of your program (how to play a game, how to save/load files, etc).

3. Explains your choice of data structures for representing Jigsaw Sudoko boards.

4. Explains how your code deals with error cases and ending.

5. Explains the additional features that you implement. You should start by listing all the additional features that you have implemented, and then explain those features and how their implementation works.

1. Describes how to build your project:

I develop my project on macOS, please use equivalent steps if you are using other OS.

* 1. Unzip the zip file, you then can get a folder.
  2. Open up a terminal and change into the project directory.

“cd ../YourPath/FinalProject-sudoku”

* 1. Lauch ghci in terminal.

“make run” or “ghci”

* 1. Type in “main” in terminal to start the game successfully.

1. Describes the functionality of your program (how to play a game, how to save/load files, etc).
2. Explains your choice of data structures for representing Jigsaw Sudoko boards.
   1. To represent a Cell in a sudoku game, I make use of the “data”:

data Cell = Cell Int Int Int Int deriving (Eq)

Four parameters in the data constructor of Cell represents

the column index (0-8 inclusively),

row index (0-8 inclusively),

block number (0-8 inclusively),

value (0-9 inclusively, 0 means the cell is empty, otherwise is filled up with the corresponding value), respectively

* 1. To represent a Jigsaw Sudoko boards, I make use of the “type” and Cell in 2.a:

Type SudokuBoard = [[Cell]]