Assignment 03 (Due: Friday, March 31, 2017, 11 : 59 : 00PM Central Time)

CSCE 322

1 Instructions

In this assignment, you will be required to write Haskell functions that simplify playing of the variation of Battleflood.

1.1 Data File Specification

An example of properly formatted file is shown in Figure 1. The encodes a game and the clusters that are to be changed during every move. Clusters are formed by 4-direction connectivity (up, down, left, right) and numbered from top-left to bottom-right in the game.

part01test01.bff

```
(
[44,43,42,41,40,38,36,35,34,32,31,30,29,27,25,24,22,17,8,4,2,1],
[
"1321311223122",
"2221231113332",
"1331231113321",
"2222232111213",
"3311331312233",
"1112332213311",
"2233111131311",
"1212331133331",
"3113123223122"
]
)
```

Figure 1: A properly formatted game encoding

2 One Player, One Move

The first part (`onePlayerOneMove` in the file `csce322h0mework03part01.hs`) will take in two (2) arguments (the game and a cluster) and returns one (1) value (the game that is the result of replacing the cluster with `1`s) . If the game has already been won, or the cluster does not exist, the game is unchanged.

```
(
[44,43,42,41,40,38,36,35,34,32,31,30,29,27,25,24,22,17,8,4,2,1],
[
"1321311223122",
"2221231113332",
"1331231113321",
"2222232111213",
"3311331312233",
"1112332213311",
"2233111131311",
"1212331133331",
"3113123223122"
]
)
```

Figure 2: Before `onePlayerOneMove`

```
"Result"
"1321311223122"
"2221231113332"
"1331231113321"
"2222232111213"
"3311331312233"
"1112332213311"
"2233111131311"
"1212331133331"
"3113123223111"
""
```

Figure 3: After `onePlayerOneMove`

3 One Player, Many Moves

The second part (`onePlayerManyMoves` in the file `csce322h0mework03part02.hs`) will take in two (2) arguments (the game and a list of clusters) and returns one (1) value (the game that is the result of replacing the clusters with `1`s in that order) . If the game has already been won the game is unchanged. If a cluster does not exist, the game is unchanged for that move.

```
(
[91,88,87,85,84,83,82,81,80,79,78,77,76,75,73,72,71,70,69,68,67,66,64,63,62,61,

[
"1114311323343",
"2423314321422",
"4414331241143",
"4221213144143",
"1334344114132",
"1142114411322",
"2433444223132",
"4433234244241",
"1122211242222",
"3311213322321",
"1311411221334",
"4142131113112",
"4321444213134",
"2411434441131",
"3124231222441"
]
)
```

Figure 4: Before `onePlayerManyMoves`

```
"Result"
"1111111111111"
"1121114111111"
"1111111141141"
"1111111144141"
"1111111114111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
"1111111111111"
""
```

Figure 5: After `onePlayerManyMoves`

4 Many Players , One Move

The third part (`manyPlayersOneMove` in the file `csce322h0mework03part03.hs`) will take in two (2) arguments (the game and a cluster) and returns one (1) value (the game that is the result of replacing the cluster with `1`s). If the game has already been won, or the cluster does not exist, the game is unchanged. This differs from the first part in that there may be more than 2 players in the game.

```
(
[36,34,33,31,29,28,27,26,25,23,22,21,20,19,18,14,10,9,7,5,4,3,2,1],
[
"3211221",
"2312213",
"2313333",
"3112221",
"2131312",
"2213212",
"2233222",
"1121231",
"2131332",
"2211332",
"3111331",
"3213122",
"1233132"
]
)
```

Figure 6: Before `manyPlayersOneMove`

```
"Result"
"3211221"
"2312213"
"2313333"
"3112221"
"2131312"
"2213212"
"2233222"
"1121231"
"2131332"
"2211332"
"3111331"
"3213122"
"1233112"
""
```

Figure 7: After `manyPlayersOneMove`

5 Many Players , Many Moves

The fourth part (`manyPlayersManyMoves` in the file `csce322h0mework03part04.hs`) will take in two (2) arguments (the game and a list of clusters) and returns one (1) value (the game that is the result of replacing the clusters with the number of the player who changes that cluster. If the game has already been won, or the cluster does not exist, the game is unchanged for a particular move and the next player makes the next move (in the case of a cluster that does not exist).

```
(
[18,16,15,6],
[
"22222121111212",
"22221222212222",
"12221111212111",
"21112212112221",
"11221222122112",
"12222112222122",
"22122122122111",
"22221211121211",
"21222111122112",
"22222112211222",
"12122121221211",
"22111121111121"
]
)
```

Figure 8: Before `manyPlayersManyMoves`

```
"Result"
"22222121111222"
"22221222212222"
"12221111212111"
"21112212112221"
"11221222122112"
"12222222222122"
"22122222122111"
"22221211121211"
"21222111122112"
"22222112211222"
"12122121221211"
"22111121111121"
""
```

Figure 9: After `manyPlayersManyMoves`

6 Naming Conventions

Your files should follow the naming convention of
`csce322h0mework03part01.hs`, `csce322h0mework03part02.hs`, `csce322h0mework03part03.hs`, and `csce322h0mework03part04.hs`.

6.1 `Helpers.hs`

A file named `Helpers.hs` has been provided with the functionality to read the `.bff` files into matrices. If a modified `Helpers.hs` file is not included with your submission, the default will be used in its place.

7 webgrader Note

Submissions will be tested with `ghc`. `cse.unl.edu` is currently running version 7.6.1 of `ghc`. If you would like to test things offline, you can load a file in `ghci` with the command `:l filename.hs` and run the main method on a given file with the command `:main "/path/to/inputfile.bff"`

8 Point Allocation

Component	Points
`csce322h0mework03part01.hs`	
Compilation	10
Test Cases	1×10
Total	20
`csce322h0mework03part02.hs`	
Compilation	10
Test Cases	1×10
Total	20
`csce322h0mework03part03.hs`	
Compilation	10
Test Cases	1×20
Total	30
`csce322h0mework03part04.hs`	
Compilation	10
Test Cases	1×20
Total	30
Total	100

9 External Resources

Learn Haskell Fast and Hard
Learn You a Haskell for Great Good!
Red Bean Software
Functional Programming Fundamentals The Haskell Cheatsheet