Presentation for Assignment 1

Implementation of Heuristic Algorithm for Board Games

April 19, 2022

Group 3

Software Modeling and Verification Chair
RWTH AACHEN UNIVERSITY

Exercise 1

map1.map 155 Bytes

```
2
2
4 1
8 8
- - 0 2 1 0 - -
- 0 0 1 2 0 0 -
i 0 0 0 0 0 0 0
0 0 0 - - 0 0 b
0 0 0 - - 0 0 0
0 0 b 0 0 x x 0
- 0 0 0 x 0 0 -
- - 0 0 0 0 - -
0 3 0 <-> 7 4 4
```

- Map Size 8*8
- 2 Players

Exercise 1

map2.map 330 Bytes

```
3
4
5 2
10 15
0 x 0 - - - 0 0 0 - - - 0 x 0
0 0 0 - - - 0 i 0 - - - 0 0 0
0 0 0 - - - 0 0 0 - - - 0 0 0
0 0 b - - - 0 0 0 - - - b 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 c 0 0 1 3 2 0 0 c 0 0 0
0 0 0 0 0 0 3 2 1 0 0 0 0 0 0
- - - - 0 0 2 1 3 0 0 - - - -
- - - - 0 0 0 0 0 0 0 - - - -
- - - - 0 0 0 c 0 0 0 - - - -
1 2 0 <-> 14 1 0
```

- Map Size 10*15
- 3 Players

Exercise 1

map3.map 1.25 KB

```
5
3
4 1
25 25
- - - - - 0 0 0 0 0 0 0 b 0 0 0 0 0 0 0 - - - - -
- - - - 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - - - -
- - - 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - - -
- - 0 0 0 x 0 0 0 0 0 0 0 0 0 0 0 0 0 x 0 0 0 - -
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 c 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0 i 0 0
0 0 0 0 0 0 0 0 0 0 0 - - - 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 - - - - - 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 1 2 3 4 5 0 0 0 0 0 0 0 0 0 -
- - 0 0 0 0 0 0 0 0 2 3 4 5 1 0 0 0 0 0 0 0 - -
- - - 0 0 0 b 0 0 0 3 4 5 1 2 0 0 0 0 c 0 0 - - -
- - 0 0 0 0 0 0 0 0 4 5 1 2 3 0 0 0 0 0 0 0 - -
- 0 0 0 0 0 0 0 0 0 5 1 2 3 4 0 0 0 0 0 0 0 0 -
0 0 0 0 0 0 0 0 0 0 - - - - - 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 - - - 0 0 0 0 b 0 0 0 0 0
0 i 0 0 0 0 0 0 0 0 0 0 - 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
- 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 -
- - 0 0 0 x 0 0 0 0 0 0 0 0 0 0 0 x 0 0 0 - -
- - - 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - - -
- - - - 0 0 0 0 0 0 0 c 0 0 0 0 0 0 0 - - - -
- - - - - 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 - - - - -
5 6 3 <-> 10 3 4
```

- Map Size 25*25
- 5 Players

Exercise 2

- Use of **2-dimensional Array** to represent the coordinate system of the map
- Array stores objects of a class **Tile**
- Use of Enumeration class **TileTypes** combining with the class **Tile** with several methods and attributes
- Use of **Hashmap** of type (x,y,direction) $\rightarrow$ Tile(Neighbour) to store the extra transitions
- First the hashmap checks if there are special transitions, then calculate with (x,y) coordinates for neighbour calculation

Exercise 3

moveValidator method checks (in that order) if:

- the tile the move is performed on exists. No –> false
- its a bomb move. Yes –> true
- tiles in any direction are turned. Yes –> true
- else –> false

Exercise 4

- Use of **Array List** to save the empty tiles with occupied neighbours and the occupied tiles
- Override stones can be placed on occupied tiles, player stones can be placed on empty tiles with at least one occupied neighbour
- Specify bomb/override for bonus fields and player number for choice fields by adding a choice variable in the **move class**

Exercise 5

Idea:

- Players that are worse than us give positive points, players that are better than us give negative points.
- We also evaluate the distance between us and the other players.
- If the player is better than us, we want a small distance so we can hopefully catch up
- If the player is worse than is, we want a large difference so they won't catch up on us.

Exercise 5

Our heuristic calculates:

- The number of tiles for each player, then creates a ranking
- We give each other player a value, in the end all those values are summarized
- The value is positive if the player is below us in the ranking and negative if the player is above us
- The basic value is the absolute difference of the number of tiles
- this difference is multiplied with a distance factor that is proportional to the difference to that opponent.
  - distance factor if the player is above us: $d_{up} = 1 - \frac{|difference(i,us)|}{totalDifference}$
  - distance factor player above us:$d_{down} = \frac{|difference(i,us)|}{totalDifference}$

Exercise 5

Formally:

- Let T_i be the number of tiles of each player, let our player number be one for simplicity
- Heuristic $H = \sum_{i>1} H_i$
- If player i is better than us:
  $H_i = -(|T_i - T_1| \cdot (1 - \frac{|T_i - T_1|}{\sum_{j>1} |T_j - T_1|}))$
- If the player is worse than us:
  $H_i = |T_i - T_1| \cdot (\frac{|T_i - T_1|}{\sum_{j>1} |T_j - T_1|})$