Genetic algorithm.

This genetic algorithm creates “players” and then improves them through selection. These “players” have genome (length of genome is set when the algorithm is initialized) consisting of numbers 0 to 31. These numbers govern “players” behavior in the following way:

0 – 11 “look” command: looks at certain tile relative to the “player” and, depending on what is in that tile, executes the gene 1 to 4 positions ahead (genome is cyclical, if its length is 32, then position following 31 is 0).

0 – look at FIRST tile UP;

1 – look at SECOND tile UP;

2 – look at THIRD tile UP;

3 – look at FIRST tile DOWN;

4 – look at SECOND tile DOWN;

5 – look at THIRD tile DOWN;

6 – look at FIRST tile RIGHT;

7 – look at SECOND tile RIGHT;

8 – look at THIRD tile RIGHT;

9 – look at FIRST tile LEFT;

10 – look at SECOND tile LEFT;

11 – look at THIRD tile LEFT;

12 – 15 “move” command: moves in certain direction. If “player” moves onto a treasure (“T”), it gets awarded points, if it moves onto the door (“D”), it gets awarded points and end current level, if it moves onto an enemy (“E”), it loses points and ends current level. After moving it executes the next gene.

12 – move UP;

13 – move DOWN;

14 – move RIGHT;

15 – move LEFT;

16-19 “attack” command: attack in certain direction. If attacked tile has an enemy(“E”), the enemy is removed and player gets awarded points. After attacking, it executes the next gene.

16 – attack UP;

17 – attack DOWN;

18 – attack RIGHT;

19 – attack LEFT;

20 -31 – executes gene n positions ahead, where n is value of this gene.

When the “player” is created it is evaluated through a certain amount (set when the algorithm is initialized) of levels. Levels are randomly generated. Later levels are bigger and contain more enemies (“E”) and treasure (“T”).

When initializing the algorithm it is possible set number of generations that are evaluated with “Training Wheels” and without. When “players” get evaluated with “Training Wheels” they get awarded additional points for moving up the Y axis (away from the start and towards the door (“D”)). This feature was added in hopes that descendants of “players” that get awarded for approaching the door (“D”) in earlier generations would be better at finding the door later on. In practice It doesn’t appear to much affect on the end result: “players” in simulations where they went through 200 generations with “wheels” and 200 without “wheels” later and in simulations where “players” went through 400 generations without “wheels” performed more or less the same.

The selection is accomplished through “Tournament Selection”. The size of tournament is set when the algorithm is initialized.