CMPT 212 (2008-1) Assignment 3 — Evaluation

Ján Maňuch
jmanuch@sfu.ca

1 Evaluation

We used 7 test files to evaluate the assignment. To compare the output produced by your program and the correct output, the program **`compare.cpp`** was used. The size of your submission (for the contest) was then determined with **`count.cpp`** (ignores comments and spaces).

2 Test files

Test 1 Game of Life (10 points) — public test 1 (agents don't move, clone nor die, just stay where they are).

Test 2 Running agents (10 points) — public test 2 (agents move).

Test 3 Cloning agents (10 points) — public test 3 (agents clone).

Test 4 2-step Game of Life (20 points) — another implementation of Game of Life using agents. The dead cells do not contain any agents, while living cells contain exactly one agent in every second step. In intermediate steps, each living cell will clone offsprings to all surrounding cells. Now, the living cell (agent) will survive only if there are 2 or 3 offsprings in its position from surrounding cells (the offsprings will die, but they will help the living cell to survive). The dead cell which contains exactly 3 offsprings will change to living cell, by changing the first offspring (with position 0 in the **`myposition`** list) to a proper living cell, while the other offsprings will die. Well, a bit cruel Game of Life, but it works (agents clone and die).

Test 5 2-step Game of Life (10 points) — this time running 100 steps.

Test 6 Priority test (10 points) — there could be many agents in one cell with different priorities. The test 6 checks (i) whether the correct state is determined for each cell; (ii) the functionality of **`remove_agent(Region)`** (no **`step()`** used, so agents do not do anything).

Test 7 Rabbit test (10 points) — agents get a random sequence (DNA) and sex (male/female) when they are created and move randomly around field based on their random sequence, producing offsprings when they meet and die when they get old (agents move, die and clone).