

Course C⁺⁺, Exercise List 12

Deadline: 13.06.2014

The *game of life* was invented by John Conway in 1970. It is a game in which figures develop by themselves in a two dimensional grid. The grid is built-up using booleans. The grid can be infinite in principle, but of course we can implement only a finite grid. At each point time t , it is computed for each boolean in the grid, what will be the state of the boolean on the next moment $t + 1$. The computation rules are as follows:

1. If a cell is on, and has fewer than two neighbours that are on, it will be off at the next moment in time.
2. If a cell is on, and has two or three neighbours that are on, it will stay on at the next moment in time.
3. If a cell is on, and has more than three neighbours that are on, it will be off at the next moment in time.
4. If a cell is off, and has exactly three neighbours that are on, it will be on at the next moment in time.

See also http://en.wikipedia.org/wiki/Conway%27s_Game_of_Life.

1. Download **grid.h**, **figure.h**, **figure.cpp**, **life.cpp**, **Makefile** from the course homepage, and complete the missing constructors and assignment operators of **class grid**. Write a destructor for **class grid**. Write the clear method. Write the << operators.
2. Write the two **addfigure()** methods.
3. Write **nextgeneration()**.
4. We are going to plot the grid. For this, we use OpenGL (<http://www.opengl.org/>) and SFML (<http://www.sfml-dev.org/>), which are installed in rooms 7 and 137.

Replace `#define USEGRAPHICS 0` by `#define USEGRAPHICS 1`.

Site www.argentum.freeseerve.co.uk/life.htm contains some more suggestions for interesting figures, if you are not satisfied with the given figures.