

User Notes for “Game of Life” (GoL)

GoL only requires user input at the start of a run. Thereafter, all moves are automatic.

To set up a run: the user can input squares (in the large square box) by using left clicks of the mouse to set up a pattern. Alternatively, there is a random setup by using the “initialise” button. There are also several pre-defined patterns given by the buttons (Gun, P60Gun, Eater, Man, Not, And, Eater2, Snark, Snark+).

To start a run: hit the “run” button. The speed of the run is controlled by the slider bar “Time between steps” in milliseconds. The information text will show as “running”.

To stop/pause a run: hit the “stop” button. The information text will show as “dormant”.

To clear a run: hit the “clear” button. This will set all squares to blank and reset the counter to zero. It does NOT stop the run.

Edges: The board (shown as the large square box) can have distinct edges where moving squares hit a “brick wall” or it can have no edges. No edges means that cells disappearing on one side will reappear on the other side. Technically this is topologically equivalent to a torus. Edges are controlled by the “edges” button! The information text will show the current setting. **NB.** Certain run types (guns, logic gates) will work only with edges on as any stray squares will destroy the pattern. Try it both ways and see.

Logic gates: [Advanced] (Ensure edges are on). These are the “Not” and “And” buttons. To demonstrate these fully, the user will need to switch the flow of gliders on and off. This is done by adding (or not adding!) so-called “eaters” which will “eat” a flow of gliders. Use “eater” and/or “eater2” buttons to add these as required. Experiment with all combinations to prove that the logic gates work.

Guns: (Ensure edges are on). The “Gun” and P60Gun” demonstrate how a flow of gliders can be produced.

Rules:[Advanced] Initially “standard” rules apply to GoL. These say that a live square stays alive if there exactly 2 or 3 neighbours who are alive. Otherwise death occurs. A dead square will become alive if it has exactly 3 neighbours alive. These are compressed into binary form on the information text (alive rules: $2^2+2^3=12$, dead rules: $2^3=8$). Other rules can be used (“carpet”, “random”, “special”). Carpet rules work best for a single square input in the screen centre, whilst random input works best for the other two. Run them and see – the binary compression of the rules will be displayed. Currently, the user cannot enter their own rules except by copying the html and amending the “SetRules()” function.

Further notes:

- 1) Mouse input only works if run is “dormant”.
- 2) It is worth watching a few pre-defined patterns before trying your own patterns.
- 3) It can be useful to set the time between steps to 1000 when learning. This enables the user to watch each step closely.
- 4) Rules can be changed during a run. This experiment works best with an initial random setup.
- 5) Most random rules produce a mess! However a few will produce quite stable and interesting patterns.