

# Conway's Game of Life for Microsoft Windows

## Contents

- 1.0 The Meaning of Life
- 2.0 The Screen Display
- 3.0 Using the Mouse
- 4.0 The File Menu
- 5.0 The Edit Menu
- 6.0 The Options Menu
- 7.0 Keyboard Commands
- 8.0 Playing the Game

---

## 1.0 The Meaning of Life

The Game of Life was developed by Cambridge mathematician John Horton Conway and popularized in an article by Martin Gardner in Scientific American in 1971.

Conway's Life is an example of "cellular automata," a field of mathematical study that has found application in image processing and computer simulations. However, the Game of Life is merely a mathematical recreation and has no practical value, except perhaps as an introduction to cellular automata.

The Game of Life takes place on a two-dimensional grid of squares that are either "alive" (filled) or "dead" (empty). Each square has eight "neighbor" squares adjoining it. From a given starting position, successive generations are computed according to the following rules:

If an empty square has exactly three filled neighbors, it will itself become filled. Otherwise, it will remain empty.

If a filled square has either two or three filled neighbors, it will remain filled. Otherwise, it will become empty.

From these simple rules, elaborate and fascinating patterns can develop. Some patterns expand, some disappear or become static, some oscillate, some appear to move purposefully.

This program allows you to create your own patterns and set them in motion on the playing field of Life.

## 2.0 The Screen Display

In this version of Conway's Game of Life, the cells are represented by colored squares against a black background. The size and color of the squares may be chosen from the Options Menu. Patterns may be created and edited interactively with the mouse (see Using the Mouse) and saved or loaded from the File Menu.

At the bottom of the display are pushbuttons for starting and stopping the game and a status window that shows the population and generation number of the current position.

The "playing field" grid on the screen wraps around; if a square moves off the bottom of the screen, it will reenter at the top, and if it moves off the left hand side, it will reappear on the right.

The size of the playing field is determined by the size of the window and the size of the squares. If the user changes either the size of the window or the size of the squares, the grid will be redrawn and any existing pattern will be centered in the new playing field. If the new grid is too small to hold the pattern on the screen, the pattern will be truncated. The user may choose (from the Options Menu) whether or not to be notified when truncation takes place. Notification is the default.

## 3.0 Using the Mouse

The left and right mouse buttons are used to create and edit patterns on the screen.

By default, clicking and releasing the **left mouse button** on a square will toggle it from empty to filled or vice versa. Clicking the left mouse button on an empty square and dragging slowly with the button held down will fill all the empty squares over which the mouse passes; this is handy for "drawing lines" of squares or filling in areas.

By default, the **right mouse button** is used for cut-and-paste operations. Clicking and holding down the right button while dragging the mouse will stretch a white rectangular highlight around an area; this is sometimes called "rubber-banding." After the button is released, the highlighted rectangle may be cut or copied to a buffer, mirrored horizontally or vertically, or rotated 90 degrees with commands from the Edit Menu. The highlighted area may also be moved or deleted with

## Keyboard Commands.

Once a rectangular area has been cut or copied to the buffer, it may be pasted back into a different location. Double-clicking the right mouse button will put a white "X" on a square to mark the upper left hand corner of the paste destination. Then pasting is accomplished with a command from the Edit Menu.

For those who don't have (or prefer not to use) a right mouse button, there is a command on the Options Menu that will reverse the functions of the mouse buttons. With the buttons reversed, the left button is used for cutting and pasting and the right button for filling or emptying individual squares.

## 4.0 The File Menu

Life patterns are saved in a simple text file format in which empty squares are represented by '.' and filled squares by '\*'. The default file extension is ".LIF". The logo displayed at startup is contained in the file GETALIFE.LIF; deleting or modifying this file will cause the program to start with a blank screen or with the pattern of your choice.

Commands on the File Menu include:

**Load...** -- Brings up a file selection dialog and loads the selected pattern centered on the screen. The previous pattern is erased and the population and generation counters are reset.

**Save** -- Saves the current screen pattern to a file. This option isn't enabled until a file name has been chosen with the Save As command.

**Save As...** -- Brings up a file selection dialog and saves the pattern currently on the screen. The default file name is "PATTERN.LIF".

Also on the File Menu, for want of a better place to put them:

**Erase All** -- Clears the screen and resets the population and generation counters.

**Center** -- Centers the current pattern on the screen.

**Exit** -- Ends Life as we know it.

## 5.0 The Edit Menu

The Edit Menu contains commands for cutting, pasting and modifying Life

patterns in much the same way that a graphics program manipulates bitmaps.

The **Cut**, **Copy**, **Horizontal Flip**, **Vertical Flip** and **Rotate 90 Degrees** commands all assume that an area of the screen has been highlighted by clicking the right mouse button and dragging the mouse (see Using the Mouse). These menu items will be disabled until a highlighted area exists.

The **Paste** command is disabled until an area has been cut or copied to the buffer and until a paste destination has been chosen by double-clicking the right mouse button (see Using the Mouse). If the rectangle in the buffer is too large to copy to the chosen paste destination, it is truncated.

The **Horizontal Flip** and **Vertical Flip** commands mirror the contents of a highlighted rectangle. They appear to work in place on the screen, but they use the buffer as a working area and modify its contents.

The **Rotate 90 Degrees** command cuts the highlighted rectangle, rotates it a quarter turn clockwise, and pastes it back on the screen at the position of the upper left corner of the original rectangle. Since it is in effect a cut and paste, it is subject to the same truncation as the Paste command if the entire rotated rectangle will not fit onto the screen.

## 6.0 The Options Menu

The Options Menu controls various aspects of the program's appearance and performance:

**Cell Size** -- A choice of cell sizes from 3 to 20 pixels is offered. The default is 7. Each cell has a one-pixel boundary between itself and its neighbors.

**Timing** -- The minimum time between generations can be set to as little as 1/10 of a second or as much as 5 seconds. The default is 1/2 second. Notice that these are minimum time delays; if each new generation takes longer than 1/10 of a second to compute, you will not see ten generations per second. Your mileage may vary.

**Color** -- The default color scheme is Rainbow, in which each cell is assigned a color of the rainbow determined by its age; a newly filled cell will be red, will turn orange in the next generation, and if it lives will proceed through the spectrum to violet. Alternately, a single color may be chosen for all cells.

**Show Grid** -- You may find it helpful when designing patterns to see the underlying checkerboard grid instead of a plain black background. Selecting this option makes the boundaries between the cells visible.

**Notify of Truncation** -- By default, the program will notify you when a change in window or cell size or the attempt to load too large a file has caused part of the pattern to be truncated. If you dislike intrusive message boxes, you may turn this option off.

**Reverse Mouse Buttons** -- Usually the right mouse button is used to highlight rectangular areas and select destination points for cut and paste (see Using the Mouse). Selecting this option will assign that function to the left mouse button instead.

## 7.0 Keyboard Commands

In addition to the usual keyboard alternatives for Windows menu commands, this program has a few commands of its own that can only be given with keystrokes:

**Space Bar** -- Pressing the Space Bar advances the screen position by one generation. This "single step" mode is useful when you wish to study how one position is transformed into another.

**Arrow Keys** -- Hitting any of the four arrow keys will move the entire pattern one cell in that direction, if possible within the current screen limits. If a rectangular area has been highlighted with the right mouse button (see Using the Mouse), only that area will move in the chosen direction. As the highlighted area moves, it will erase anything in its path.

**Delete** -- Hitting the Delete key while a rectangular area is highlighted will erase all the cells in that area. This is different from the Cut command on the Edit Menu because the buffer isn't affected by the Delete.

## 8.0 Playing the Game

John Conway has called his Game of Life a "no-player game"; once a pattern has been created and set in motion, it proceeds toward its inevitable fate without outside intervention. The enjoyment we get from Conway's Life is largely aesthetic, as we marvel at the way patterns evolve and change.

However, if you have a competitive bent, you can play the Game of Life with the following object: while a pattern is going through its generations on the screen, try to make it disappear completely by clicking the left mouse button to add cells in the right places. To support this mode of play, the program will not allow you to delete individual cells while the game is in progress; that would make it too easy.

As a computer game it's not likely to replace chess or tetris, but that's Life!