Langton's Ant
programmed by Alexander M Kasprzyk

Program Information

"Langton's Ant" will work on any Macintosh with Colour QuickDraw and needs the Thread Manager.

Please note that although I, Alexander M Kasprzyk, programed "Langton's Ant" the idea for the rule–set was not mine, it was invented by Christopher Langton.

I put a lot of effort into writing this program, but it is Freeware. If you like and use the program I wouldn't mind an eMail to kasprzyk@cityscape.co.uk


Background Information

One of the simplest cellular automata is "Langton's Ant", named after its inventor, Christopher Langton. The "ant" effects the state of the world each iteration by moving forwards one square.
- If it is currently on a black cell the ant turns 90° to the left
- If it is currently on a white cell the ant turns 90° to the right
- As it moves to the next square, the one that it is on changes colour from white to black, or visa versa

On the face of it, Langton's ant would seem to be a simple animal – after all the rules are less than complex. In fact the ant displays behaviour that is currently baffling mathematicians.

Suppose you start the ant in a northwards direction on a white grid; the first move will turn the ant right so that it is facing south and take it forwards one square, turning the starting square black. Because it is now on a white square the ant will turn right so that it is now facing west and then moves it onto another white square, turning the last square black. After a few moves the ant will start revisiting earlier squares that have turned black. Very quickly the ant's movements become quite complicated *(see fig.1)*.

(fig 1) Langton's Ant after 12 iterations
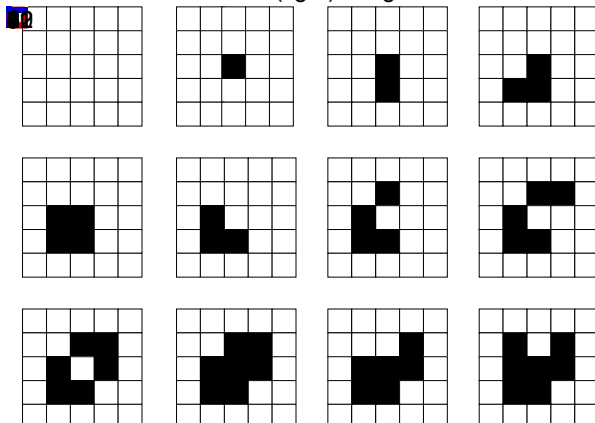
Every so often during the first few hundred moves the ant produces a nice symmetrical pattern. After this things get rather chaotic for a few thousand moves, then something amazing can happen; the ant can become locked into a cycle that repeats the same sequence of 104 moves, the overall result being to move the ant two squares diagonally down towards the left. It continues like this forever (or until the ant encounters some previous trail), systematically building a broad diagonal "highway".

Fig.2 shows two parallel highways constructed by two ants.

(fig 2) Two "highways" being constructed in the bottom left corner

This behaviour is interesting, but experiments show that if you scatter any number of black squares around before the ant sets off, it still ends up building the highway. The problem baffling mathematicians is that nobody can provide proof that the ant always ends up building a highway for every possible configuration of black squares, though it certainly seems that it does.

Using the Program

When you load up "Langton's Ant" and select **New** from the **File** menu you will be presented with an empty window. This is the world on which you can place as many ants as you wish. You can also change the colour of any tiles you want.

Select the **Add Ant** item from under the **Simulation** menu. Now click and hold anywhere in the window – this allows you to add an ant onto the world. While you hold down the mouse button you can move the mouse round to change the direction the ant will face in. You can add as many ants as you want like this. (Note: You can not start a simulation if the world does not contain any ants.)

Now select **Invert Tile** from the **Simulation** menu. Click in the window – the colour of the tile you have clicked on will invert, i.e., if it is a white tile it will turn black, or if black it will turn white. If you hold down the mouse button and drag you can colour in

multiple tiles with the colour you started with.

To start the simulation select **Continue** from the **Simulation** menu.

To end the simulation select **Close** from the **File** menu, or click in the window's close box.

Multiple simulations can be run simultaneously.

Ants can only be added to a simulation and tile colours inverted when the simulation is paused. To do this select **Pause** from the **Simulation** menu.

When you are totally finished select **Quite** from the **File** menu to exit the program.

Have fun!