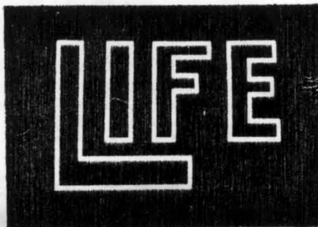
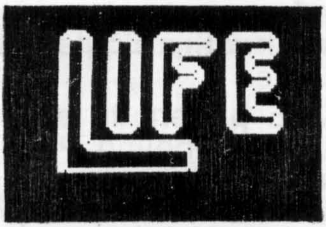
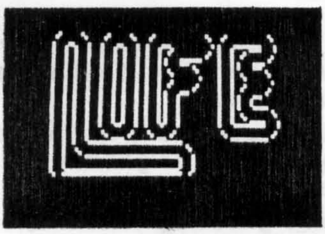


FIRST GENERATION



SECOND GENERATION



THIRD GENERATION

Distributed by George Moses Company
P.O. BOX 686, Brighton, MI 48116
$16.00 U.S. Funds

LIFE

© Jay Fenton, 1982

LIFE is a machine language program that is loaded with **:INPUT GO.** A short program will come up on the screen first that will demonstrate the functions of the keypad in the LIFE generation controls. After noting the positions of the four control keys you will see the screen go black with a single white square in the screen center.

The initial LIFE pattern is entered using joystick 1 just as in Scribbling. Pull the trigger to write or erase. Write or erase mode is selected by the knob. Turn the knob right to erase or left to write.

Remember the following rules as you place your first generation pattern of "cells" around the screen:

- ***If a live cell is surrounded by two or three live cells in the present generation, it will remain on (or live) in the next generation.***
- ***If an empty cell is surrounded in the present generation by exactly three neighbors, the cell will be on (ie: born) in the next generation.***
- ***If a cell has no neighboring live cells, or only one neighbor, it dies of loneliness and will be turned off (die) in the next generation.***
- ***If a cell has four or more live cells neighboring it, it will die in the next generation from overcrowding.***

These rules are to be applied simultaneously to every cell in the pattern. The application of the rules to every bit in the field constitutes a generation.

David J Buckingham
Computer Communications Network Group
E4, room 2369A
University of Waterloo
Waterloo, Ontario
CANADA N2L 3G1

Introduction

Life is a game that was developed by Prof John H Conway at the University of Cambridge and first presented by Martin Gardner in the October 1970 "Mathematical Games" column in *Scientific American*. The game is derived from a field of mathematics known as *automata theory* (in this case *cellular automata*). In the February 1971 "Mathematical Games" column the game was described again along with a good introduction to automata theory.

The game is played on a uniform cellular grid (in this case an area divided into squares, such as graph paper) where every cell is surrounded by eight immediate neighbors (ie: cells touching the center cell under consideration). Each cell, or *automaton*, can be in either a 1 or 0 state (on or off – alive or dead). The population of cells is changed by a set of predetermined rules. These changes proceed in intervals called *generations*.

The rules are as follows:

- If a live cell is surrounded by two or three live cells in the present generation, it will remain on (or live) in the next generation.
- If an empty cell is surrounded in the present generation by exactly three neighbors, the cell will be on (ie: born) in the next generation.
- If a cell has no neighboring live cells, or only one neighbor, it dies of loneliness and will be turned off in the next generation.
- If a cell has four or more live cells neighboring it, it will die in the next generation from overcrowding.

These rules are to be applied simultaneously to every cell in the pattern. The application of the rules to every bit in the field constitutes a generation. See figure 1 for an example of rule applications.

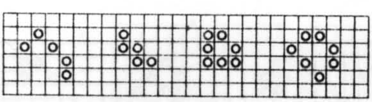


Figure 1: Transformation of a Life pattern through three generations. This process is sometimes referred to as the automation of the pattern. Generation 0 is the original pattern of live cells. The succeeding generations proceed according to the rules of birth, existence and death.

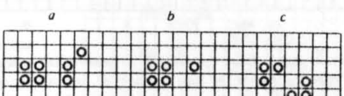


Figure 2: Example of object contiguity. Group a is considered to be two distinct objects; groups b and c are considered to be single objects.

Unresolved Questions

What is a unique object in this universe of cells? What is a collection of objects? How do we tell them apart? These are difficult questions to answer conclusively. For the purposes of this article, an object is a cluster of connected bits or cells, a collection of clusters which will cause births by being near one another, or a collection of clusters that prevent some birth that would otherwise occur. Figure 2 gives some examples of patterns that would be objects and some that would not.

A collection of distinct objects is referred to as a *constellation*. Some constellations are so common that they are named as though they were a single object. Some of these are presented in figure 3.

Objects

Most people with access to some sort of computer have probably had a chance to observe the variety of patterns that exist within Life and to note some of the special properties particular to some of these objects. In order to be able to manipulate these objects, they have been classified.

The major groupings of classification are still lifes, oscillators, spaceships, uniform propagators, and a catch-all group of random objects. A rough outline of this system is shown in table 1. I shall attempt to describe

	Subclass	Number of objects known	Smallest object(s)
Class I (still lifes)	subdivided by symmetry	∞	block
Class II (oscillators)	(IIa1) flip flops (IIa2) on-offs	∞	blinker beacon
	(IIb) billiard table configurations	>100	MIT oscillator
	(IIc) inductors	4	tumbler
	(IId) pulsators	8	mazing, pentadecathlon
	(IIe) shuttles	5	shuttle
	(IIf) eater bound	23	two eaters
Class III (spaceships)	(IIIa) diagonal	1	glider
	(IIIb) orthogonal	3	lightweight spaceship (LWSS)
Class IV (uniform propagators)	(IVa) stationary	2	glider gun
	(IVb) moving (puffer trains)	4 types	switch engine
Class V (random)	subdivided by type of objects in census	∞	bit (single cell)

Table 1: Classes and subclasses of objects occurring in Life, along with supplementary information.

Number of live cells	Number of still life patterns
1	0
2	0
3	0
4	2
5	1
6	5
7	4
8	9
9	10
10	25
11	46
12	121
13	240
14	619

Table 2: The number of small still life patterns which occur for each number of live cells up to 14.

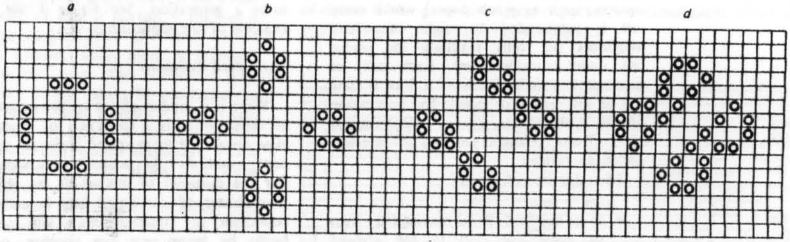


Figure 3: Commonly occurring constellations. These are not a single object, but bear names for convenience, as follows: a, traffic light; b, honeyfarm; c, fleet; d, bakery.

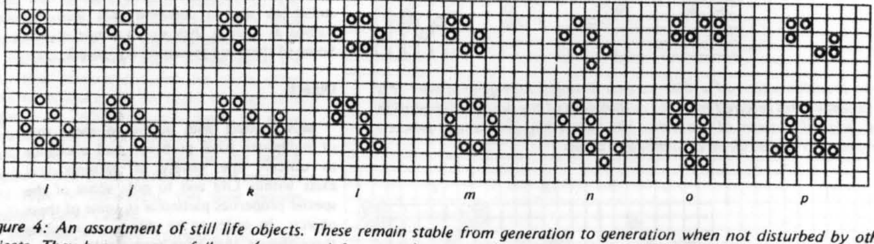


Figure 4: An assortment of still life objects. These remain stable from generation to generation when not disturbed by other objects. They bear names as follows: (top row, left to right) a, block; b, tub; c, boat; d, beehive; e, ship; f, barge; g, snake; h, aircraft carrier; (bottom row, left to right) i, burloaf; j, long boat; k, long snake; l, period 3 eater; m, pond; n, long barge; o, shillelagh; p, hat.

each class and some of the objects of particular note within each class.

Class I: Still Lifes

Still lifes are objects in which there are no births or deaths and so remain the same from generation to generation. These particular objects are fairly easy to enumerate. An associate of mine, Peter Raynham, wrote a program which found all still lifes of less than 15 bits. The statistics of their distribution are presented in table 2. Some of the smaller ones are shown in figure 4.

One of the most practical uses of a still life is as an *eater*. An eater is an object capable of destroying or modifying another object and being able to return to its original configuration. Still lifes are good for this since they are able to attack any configuration at any phase (they are period 1 objects and do not change).

At present we know of three different eaters, each able to attack different types of objects. By differing objects, I mean objects that have different border configurations. Since the eater attacks only the outside surface of an object, this outer surface determines which type of eater might be suitable for use. Each eater will be described with an object that it can "eat" to show how that eater works.

The smallest member of the eater family is the *block*, shown in figure 4a. The block is effective in consuming objects that have one connected bit in the row facing it (as in figure 5). Its other reason for utility is the fact that it is very small. In oscillators and spaceship guns, where there may be little room for the removal of spurious debris, there is usually enough room for a block.

The second object of this family is the most important and versatile. It is the 7 bit still life referred to in figure 4l as the period 3 eater. This object attacks other objects that exhibit a flat connected outer border that is at least two bits long. Figure 6 shows such an attack. Almost all objects will develop this type of border if they expand. This property renders the period 3 eater invaluable. Although it is not quite as small as the block, it is still very much smaller than any other of the eater family.

Note:

A bibliography containing all *Scientific American* Life articles referred to in text is provided at the end of this article.

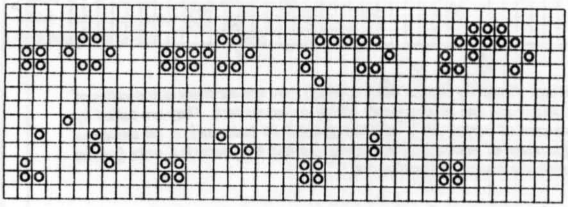


Figure 5: A block devours a beehive. This process requires seven generations, as shown here.

The third such object, the period 6 eater, exhibits similarities to the period 3 eater in the way it eats; however it requires six generations to dispose of its prey and return to its initial state, whereas the previous eater takes three generations. This increase in time is important for success if the object being eaten has left some transient debris near the eater. If the eating mechanism were to reform itself quickly, this debris could kill the eater. In this case, the eater does not reform for an extra three generations, during which time this debris may well vanish.

Most of the period 6 eater's prey are the same as the period 3 eater's; but both are able to attack certain additional objects, complementing each other very nicely. Figure 7 shows the period 6 eater conveniently disposing of a block.

Class II: Oscillators

Oscillators are nonmoving objects with periods of two and greater. A *blinker*, shown in figure 8a, is a simple oscillator consisting of three cells alternating in subsequent generations between a vertical and horizontal row. At present, we know of roughly 150 unique oscillators with a period greater than two.

There is a large undetermined number of period 2 oscillators, since they are very easy to construct. The oscillators have been subclassified by relating their mechanisms and their degree of naturalness. (Natural objects are those which may evolve from random patterns of live cells without intervention by the experimenter.)

Since there are only two basic ways in which a period 2 oscillator can work, these objects are very well defined. Therefore, they are assigned to their own subclass (class IIa). They must work as *flip flops*, *on-offs* or a combination of the two. In a flip flop, deaths occur because of *under*population. In an on-off any deaths that occur are due to *over*population. (This is almost always true. Figure 9 is a period 2 oscillator, which if traced, will reveal that it adheres to both definitions.) A variety of small period 2 oscillators is shown in figure 8; the type of each oscillator is also given.

Next in the hierarchy are *billiard table*

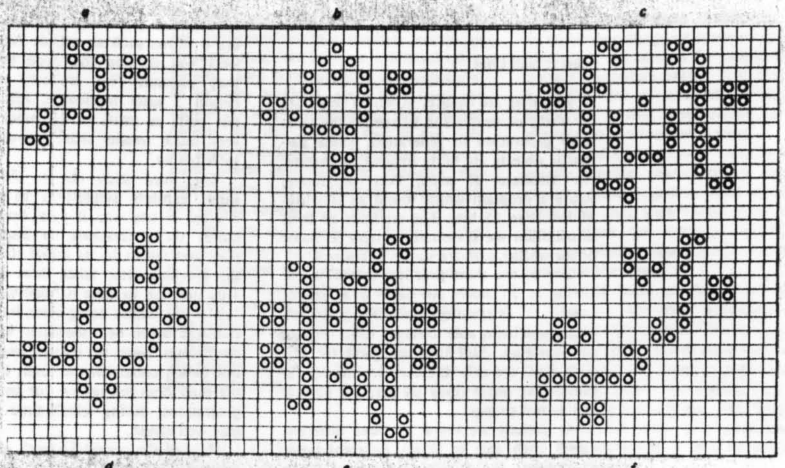


Figure 10: Billiard table configurations. These oscillate within an enclosed area, as do balls on a billiard table. These are artificial objects which have not occurred unless specifically constructed by the experimenter. They tend to be large; the smallest is composed of 18 live cells. Those illustrated bear names as follows: a, MIT oscillator (a period 3 object); b, burloaferimeter (period 7); c, an unnamed period 8 object; d, wavefront (period 4); e, an unnamed period 5 oscillator; f, an unnamed period 9 object.

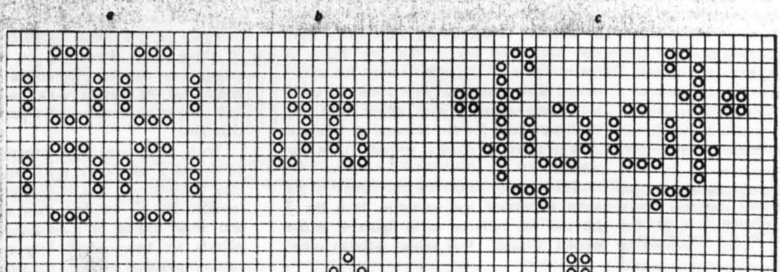
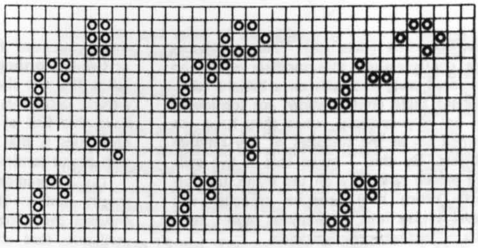


Figure 11: Inductor and pulsator oscillators. These are natural objects which may appear from automation of random patterns. Inductors possess an imaginary line of symmetry which pulsators lack. They are called by the following names: a, pulsar (an inductor of period 3); b, tumbler (period 14 inductor); c, an unnamed period 8 inductor; d, pentadecathlon (pulsator of period 15); e, mazing (period 4 pulsator); f, unix (period 6 pulsator). The pentadecathlon is of particular historical significance.

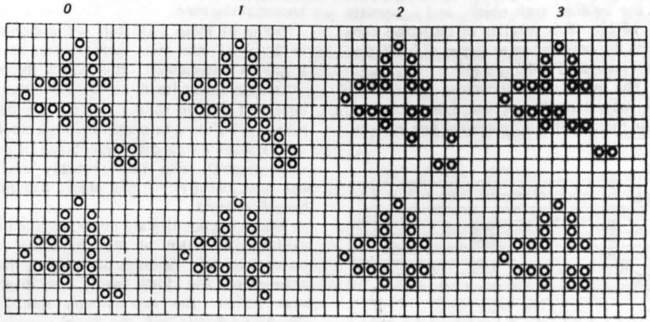
0 1 2



3 4 5

Figure 6: The most versatile eater object, the period 3 eater, devours the precursor pattern to a beehive. While in isolation a 7 cell still life, this eater attacks other objects with a flat connected outer border at least two cells in length.

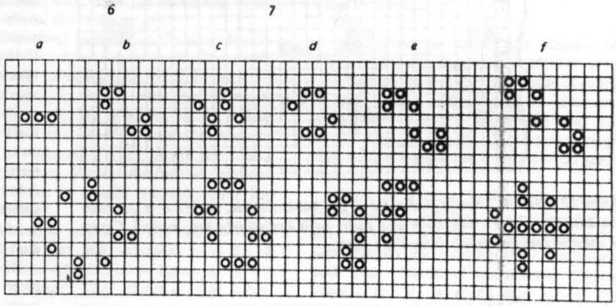
0 1 2 3



4 5 6 7

Figure 7: A period 6 eater attacks and eats a block. This new eater is notably more symmetrical than the period 3 eater. Functionally, each complements the capability of the other.

a b c d e f



g h i j

Figure 8: A variety of small period 2 oscillators. These objects alternate in succeeding generations between two patterns. Those on the top row bear names as follows (left to right): a, blinker; b, beacon; c, clock; d, toad; e, bipole; f, tripole; those on the bottom row are unnamed flip flops. The beacon is the only example here of an on-off type oscillator.

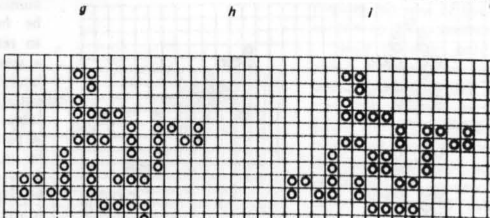


Figure 9: A period 2 oscillator which functions both as a flip flop and as an on-off.

configurations (class IIb). These oscillators are configurations that oscillate within an enclosed area, like balls on a billiard table. Billiard table configurations are considered to be very artificial, since they have not turned up in the histories of any random objects. By this, I mean that if live cells are placed randomly on the plane, the patterns which they generate probably will never evolve into an artificial object, such as a billiard table configuration.

They are quite large, as evidenced by the examples in figure 10. The first example is the smallest such object, and it consists of 18 bits. This subclass of oscillators contains the only known examples of oscillators with periods of 7, 10 and 11.

The next class, *inductors* (class 11c), are natural oscillators that exist in two or four pieces with an imaginary line of symmetry between them (exhibiting one-way or two-way orthogonal symmetry). *Pulsators* (class IId) are also so far considered to be natural oscillators except that they do not have this line of symmetry. One of their properties is that they require no external stimulus to continue oscillating.

The aforementioned subclasses have greatly similar characteristics, so I have grouped them together. Most of the initial oscillators that were found were from this group, since the methods for harnessing random objects into oscillators were not known at the time.

Some of these oscillators are presented in figure 11; the most important of these is the *pentadecathlon*. This object throws off several sparks (small collections of dying bits) that can be used to reflect a glider, reflect two gliders, turn a glider into a block, turn a block into a glider, etc. Some of the early research into Life probably might not have occurred if this object had not been discovered.

Shuttles (class IIe) are important for the existence of much of the interesting research into Life. Shuttles are objects that move back and forth with a relatively large period. The two primary shuttles, the *basic shuttle* and the more complex *twin bees*, leave debris at their extremities which would fatally wound the shuttles if the debris were not removed before they returned (see figure 12). This is one of the uses of the eaters that was discussed in the section on still lifes. In the examples I have used blocks to remove the debris from the ends, but just about any of the eaters would have suited some phase of this debris. The debris left behind may at first seem to be somewhat of a bother, but without it there would most likely not be any known glider guns (defined later).

The very last class (class IIg) contains *eater bound oscillators*. These oscillators consist of patterns which generally must be manipulated in order for the object to return to its initial state. In figure 13 a good example of two eater bound oscillators is given that also shows the differentiation between two eaters acting on the same object (which is not often possible). A period 52 oscillator (figure 14) is shown to illustrate the unusual properties of objects being eaten. The center object will be attacked by one eater twice each time it rotates (the object rotates 90° every 13 generations). The example in figure 15 is a period 6 oscillator using the period 6 eater. The 7 bit eater is not suitable here because it would have returned to its original state too soon and would have attacked the reforming object. (If the 7 bit eater

a
b

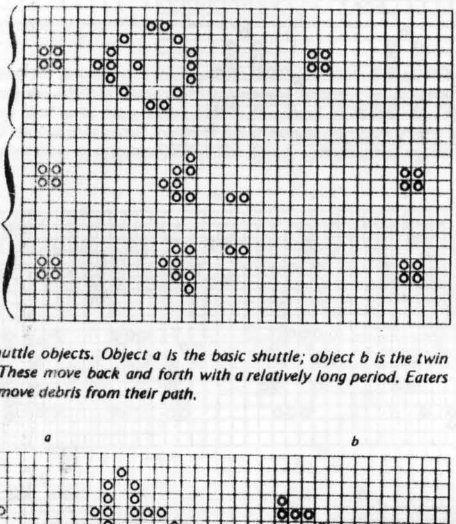


Figure 12: Shuttle objects. Object a is the basic shuttle; object b is the twin bees shuttle. These move back and forth with a relatively long period. Eaters are used to remove debris from their path.

a b

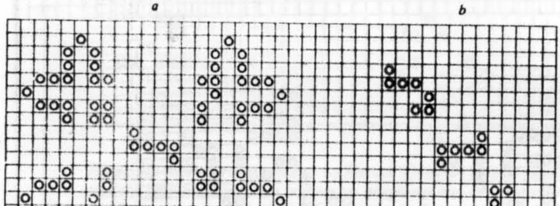


Figure 13: Two eater bound oscillators. These differ in that they are stabilized by two different eaters. Oscillator a has a period of 6; oscillator b has a period of 5.

is used, the patterns results in two blinkers, six blocks, and one tub in 110 generations.)

Class III: Spaceships

Regrettably, there have been no new spaceships reported since the orthogonal spaceships presented in *Scientific American* in 1971. These are summarized in figure 16. The *glider* (figure 16a), which features diagonal movement, has been used for many simulations and constructions.

Movement by an object of one space in one generation is referred to as movement at the speed of light (c). There is no distinction made between diagonal and orthogonal movement, even though algebraically the distances are not the same. The glider travels at c/4 and the three other spaceships travel at c/2. The interesting thing to note is that the three larger spaceships travel orthogonally. The orthogonal spaceships are most useful in several of the types of puffer trains to be discussed in the next section.

Class IV: Pattern Producing Mechanisms

Class IV is divided into two sections: the first contains static patterns that produce moving progeny, and the second contains moving patterns that produce some type of stationary or moving output.

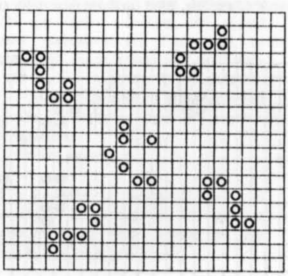


Figure 14: A long period eater bound oscillator. This object has a period of 52; 13 generations are required for 90° of rotation. The central section is attacked twice by one eater each time it rotates.

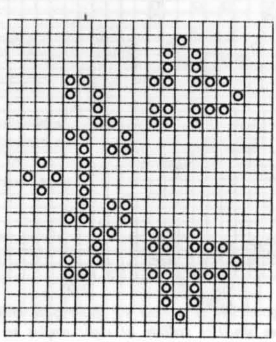


Figure 15: A period 6 oscillator which employs the period 6 eater. This matching of period frequencies prevents the eater from disrupting the reforming central group.

Spaceship Guns

Class IVa consists of *spaceship guns*. These objects eject projectiles of class III objects. The main two objects of class IVa are the *glider guns* of primary period 30 and 46. There are no primary guns which produce any of the other three spaceships. However, such a mechanism can be built using glider guns.

The period 30 glider gun (figure 17) works by having two shuttles of the type presented earlier aimed at one another. The debris that would normally be removed by eaters collides and just happens to create a glider that escapes without harming the shuttles. The period 30 glider gun is of paramount importance to simulations in Life and the possible existence of computing mechanisms. These implications will be discussed in a later article.

The period 46 gun, known as a *newgun*, also works by having two shuttles collide. It may be seen in figure 18. In this case the shuttle consists of two B heptominos (described later) travelling opposite one another to produce debris at both ends of travel. A glider is produced when these two shuttles, which are of the twin bees type, collide at right angles. There are other arrangements of this shuttle that produce gliders in other ways, including an ambidextrous variety.

There is another interesting variation: if one of the debris removing blocks is removed from the end of one of the twin bee shuttles, the gun will still work.

Puffer Trains

Puffer trains are patterns that move and leave debris in their wake. Because these patterns do move, as opposed to the stationary spaceship guns, they are not only able to produce moving debris but also trails consisting of stationary objects. Leaving stable objects is useful when the intention is to produce a train of puffers to build some sort of construction on the fly.

The three basic puffer trains all work by different means. The train which was discovered first is presented in figure 19. The center object is a *pre-B heptomino*, which, if traced, will seem to move forward until the debris in the back of it stops the uniform forward motion. In this case, the B heptomino is bounded by two lightweight spaceships able to control the object; the whole configuration puffs along at c/2.

This object reaches a stable period of 140 after a startup time of over 1000 generations. Additional spaceships may be added to the end of the object to further adjust the output from it in order to reduce the final period, the startup time necessary to reach a stable period and to adjust the output to blocks, gliders, etc.

A type of puffer similar to the previous one is called a *Schick ship* (after its discoverer). This is an interesting object (consult figure 20) in that the "engine" is really a tagalong, an object capable of being pulled along behind another object (usually a spaceship). Here, a heptomino follows a pair of mirrored spaceships. It is quite remarkable that this configuration leaves a small trail of debris behind it and that, although this debris would die if left alone, additional spaceships following behind are able to trigger the debris into varying forms of static debris. The static debris can be left behind and used later. It is relatively useful for building armadas because of the relative simplicity of creating this object from gliders (producing a basic ship requires 11 gliders).

The last type of puffer train is the smallest, a mere 11 bits at startup – the size is somewhat larger when the final repeat cycle is known, since there is transient debris in the field. This particular train travels very slowly, taking 96 generations to traverse eight spaces (speed c/12). It is also very unusual in that it is the only known puffer train that travels diagonally – the same direction as the glider, but three times as slow.

Unlike the other puffer engines, this train does not require that any other spaceship exist to bound it. To stabilize the basic engine, a block must be placed somewhere in the debris produced by the object to prevent the debris from destroying it. If the engine is run without a stabilizing block, some rising debris finally catches the engine after 11 full cycles and destroys it. The remaining field settles down to a final census only after 3911 generations!

Pertinent to the above paragraph is the fact that random patterns are quite often able to produce certain types of edge configurations, which enable them to surge forward with a great burst of speed for short periods of time. In the case of the *switch engine*, when some random exhaust manages this type of movement, this slow moving engine is easily caught.

The switch engine (presented in figure 21) will produce, after its startup time, eight blocks every 288 generations. Other debris can be produced, including gliders. Since this train travels so slowly, there are presently no real uses for it.

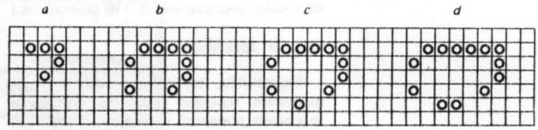


Figure 16: The four known spaceships which occur in Life. Their appellations are: a, glider; b, lightweight spaceship; c, middleweight spaceship; and d, heavyweight spaceship. The glider travels diagonally at a rate of one space every four generations. The other three travel orthogonally at one space per two generations.

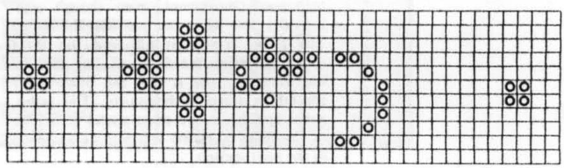


Figure 17: This glider gun, which has a period of 30 generations, was the first object of class IVa to be discovered. It periodically emits a glider which travels away diagonally. The four block still lifes are used as eaters to dispose of debris. Glider guns are of great importance in simulations, where gliders are made to collide, thus forming new objects.

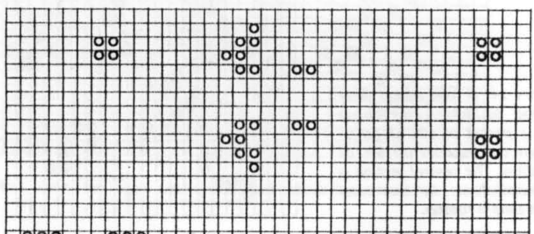


Figure 18: A period 46 glider gun which is called the newgun. Two twin bees shuttles collide at right angles to produce one glider every 46 generations. As before, the block still lifes are used to remove debris which could cause disruption of the formation.

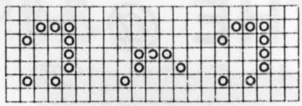


Figure 19: A puffer train constructed from the precursor to a B heptomino and two lightweight spaceships. After a startup period of over 1000 generations, it stabilizes into a period of 140 generations.

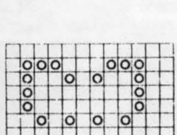


Figure 20: A Schick ship puffer train. The engine in this object is pulled along behind another object. In this example, a heptomino tags along behind a pair of mirrored spaceships. This object has a period of 12.

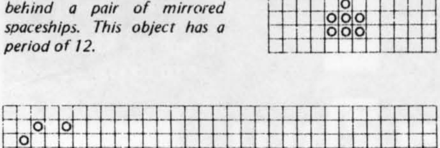


Figure 21: The switch engine puffer train, with a basic period of 96. After startup, it produces eight blocks every 288 generations after the initial stabilization of debris. Although much various debris may be created with this train, its slow speed limits its usefulness.

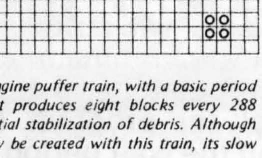


Figure 22: Several common nonterminal random objects. These are designated as follows: a, R pentomino; b, B heptomino; c, π heptomino; and d, acorn. The acorn is a Methuselah type object. Finding all the descendants of such an object is a difficult challenge.

Class V: Random Objects

A *random object* is simply anything that does not fit in any of the above classes. It appears that all random objects eventually become something from one of the above classes. It has been assumed that there are no objects that expand *irregularly* forever (this is a common problem in other cellular spaces using other rules). There are some very popular *nonterminals* in life, which, due to their commonality, have been given names. In some cases these have been rather heavily investigated. In figure 22 are some of the more common nonterminals and their names.

The most common object in this class must be the oft publicized *R pentomino* (figure 22a), which many people still believe runs forever. The result of this pattern was, however, published in *Scientific American*; it runs for 1103 generations producing four blinkers, eight blocks, one boat, four beehives, one ship, one bun leaf, and six gliders.

The *B heptomino* (figure 22b), with a census of three blocks, one ship and two gliders in 148 generations, is one of the more heavily investigated objects, as is evidenced by some of the material presented in this article. It has the following interesting property: the front configuration of the object moves along to reappear the same every other generation, but flipped over.

A close relative of the previous object is the *π heptomino* (figure 22c) with a census of five blinkers, six blocks and two ponds in 173 generations. Phase 1 of this object was called a *blasting cap* by the artificial intelligence researchers at Massachusetts Institute of Technology (MIT); we call phase 3 a *house*. If you trace the house for 30 generations, you will notice that a house reappears at the front of the debris ten spaces ahead of where it started. The house does not appear again after this because the debris catches up with it and kills it. Many attempts have been made to stabilize this object, with no success as yet.

A random object that consists of fewer than ten bits and that has descendants enduring for more than 50 generations is referred to as a *Methuselah*. The acorn pattern (figure 22d) is presently the record holder for duration. This is presented as a challenge to anyone who would like a difficult object to trace.

We hope that some of our investigations into the more exotic corners of Life will inspire readers to try their hands at this fascinating pastime.■

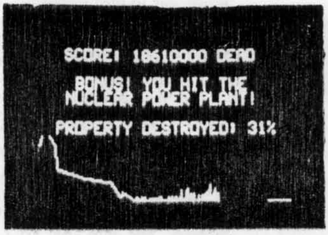
About the Author

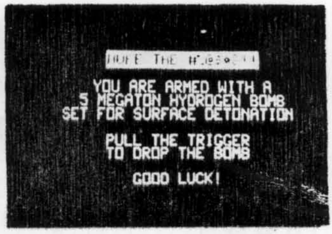
David Buckingham is an undergraduate science student at the University of Waterloo. He made a number of contributions to the now defunct publication Lifeline. *Most of these had to do with oscillators, which constitute his main field of interest in Life. He has at present found over 100 oscillators of periods greater than two.*

Buckingham's most productive area of research has been the devising of glider collisions to produce objects of classes I thru IV. As of August 1978, he has managed to create collisions to produce all of the presently known 1105 objects of less than 15 bits in size.

NUKE THE @&#%$★

© Jay Fenton, 1982





Nuke the *@&#%$*★ will automatically start after loading with the command: **:INPUT GO.** Use hand control trigger one to drop the bomb and to "speed up" the program. (Pulling the trigger will shorten the wait period for the title and score frames).

After Arcade power-up you should **RESET** without BASIC inserted before loading Nuke the *@&#%$*★. This will set the alternate color map registers which are used to display a "fallout" pattern.

The object of this game is prevent nuclear war by showing the emptiness of life even if you win the battle, because you have no human organisms left to share the victory with. So, if no nuclear war breaks out in the next five years, please give the credit to this program and its creator, Jay Fenton.

In the meantime, have fun and remember... aim for the nuclear power plant for the most devastation for your bomb dollar!!!