AMSTRAD AND SP

LANDSCA

Fill in the background to that text-only adventure. Artist Brian James puts pixel to phosphor.

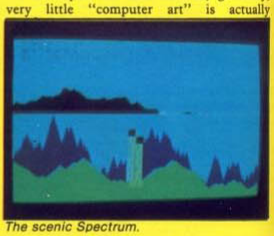
Program 1.

```
10 '       Fourier Synthesis
20 '
30 '         AMSTRAD BASIC
40 '
50 '      Brian James ,  March 1985
60 '
70 c$=CHR$(13)
90 KEY 135,"mode 1:ink 0,1:ink 1,24"+c$+" list "
100 KEY 137," edit ": KEY 128," goto "
120 SPEED KEY 12,2
130 SPEED WRITE 1
170 '
180 DIM a(250),b(250),c(250)
185 '
190 MODE 0        ' for 16 colours
200 nn=6          ' number of terms
210 am=30         ' maximum amplitude
220 INK 0,11      ' blue sky
230 '
240 FOR i= 1 TO 15
250 INK i,RND*26    ' set colours
260 NEXT
270 '
280 FOR n= 1 TO nn ' Each wavelength
290 a(n)=RND*80/n   ' amplitudes
300 b(n)=RND*2*PI   ' phases
310 c(n)= RND*15     ' colours
320 NEXT
325 PRINT: PRINT" Fourier Synthesis"
330 '
340 FOR x=0 TO 639 STEP 4
350 PLOT x,0,0       ' at bottom
360 y=10
370 k=2*PI*x/640
380 '
390 FOR n= 1 TO nn   'incr. frequencies
400 y=y+a(n)*(1+SIN(n*k+b(n)))  'add waves
410 DRAW x,y,c(n)    ' draw upwards
420 NEXT n
430 '
440 NEXT x           ' go right
450 '
460 FOR i= 1 TO 3000:NEXT   'pause
470 RUN
500 '
6000 WHILE 1:SAVE "":WEND
```

THE LANDSCAPE Creator is a program which creates detailed coloured landscape views, quickly and spontaneously. I developed this idea first on the Spectrum 48K, and now the program has been redesigned to fully exploit the beautiful graphics capability of the Amstrad CPC-464.

It is a simulation of "creativity": the computer makes all decisions relating to a wide range of parameters concerning the hills, trees, flowers, lakes, islands, and buildings. The number of variable parameters is such that the resulting scene gives endless, unexpected surprises with a wide range of possible colour schemes.

I developed this idea because, generally, very little "computer art" is actually



The scenic Spectrum.

generated by the computer. In nearly all cases the computer merely displays the picture, which is arranged by the programmer. We now have graphics packages which make it much easier for the operator to control the results on the screen. However, the result could just as well — or even better — be done with paint on canvas.

It is a much more interesting challenge to get the machine to create its own pictures, based on a "knowledge" of the subject to be portrayed. This involves intricate mathematical modelling, and really begins to unleash the power and the intelligence of the microchip and do things in a way only the computer can.

We have all seen some impressive colour graphics in computer games. In most cases these graphics make extensive use of user defined characters, which are rectangular elements — usually 8 by 8 pixels — used as building bricks. In contrast to this, the use of mathematical functions and probability theory allows us enormously improved flexibility, enabling the spontaneous creation of different scenes, giving endless surprises, even to the programmer!

It is easy enough to write a program to rearrange a number of graphic shapes — say to

CTRUM
PES

select a building from a choice of five, choose a mountain from a choice of five, etc. etc. and put them together. But to simulate the idea of "creativity", we need to use a whole hierarchy of routines, building up the picture from the smallest elements, with freedom of choice at every stage in the process. User-defined characters are of limited use in this kind of exercise.

My objective in the Landscape Creator was to have the minimum of fixed quantities: to allow the greatest freedom for variations, without departing from some basic properties of landscapes. This same idea I used in Country Cottages, a fun game for two players where you buy cottages and try your skill in managing the awful tenants.

The cottages, the landscapes and the tenants are different every time you play the game. However, the Landscape Creator was designed from scratch, to produce greater variations, better colour schemes, better appearance of water and vegetation.

The number of calculations involved is quite horrendous. Just doing a single pixel involves over 50 machine-code instructions. Though Amstrad Basic is very comprehensive and quite fast, the Landscape Creator has so many calculations to do that machine code was essential.

Routines for plotting

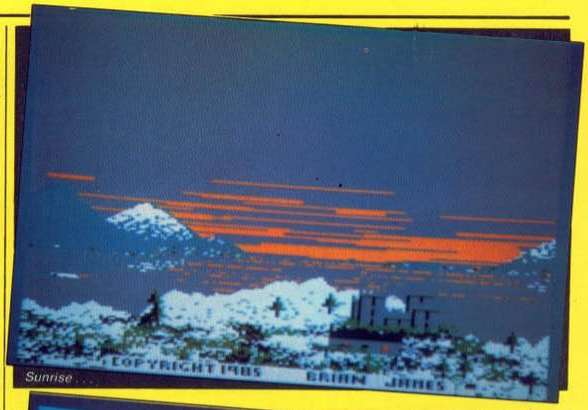
The Amstrad Firmware manual gives the addresses of inbuilt machine-code routines for plotting. These are easy to use and nicely crashproofed — but are therefore not as fast as would be desired. The Landscape Creator utilises its own extremely fast, compact plotting routines. Also the basic arithmetic routines were designed for the job.

In the Landscape Creator, the object was to have the maximum variety of pictures. Whether it is creating a rugged skyline, an island, a patch of buttercups, or a castle, a flexible routine is used which combines probability functions with appropriate mathematics.

The program needs some "knowledge" about the structure of landscapes, so that the different features will have reasonable shapes and fit sensibly together. Trees and buildings must not hang in mid-air; hills must not have enormous holes through them; land must not be drawn when it is below water! Foreground objects may hide more distant objects — not the reverse — and so on. Distant hills could have a grey or bluish colour; whereas the foreground can have much brighter colours.

Lakes and sea can use a mixture of colours

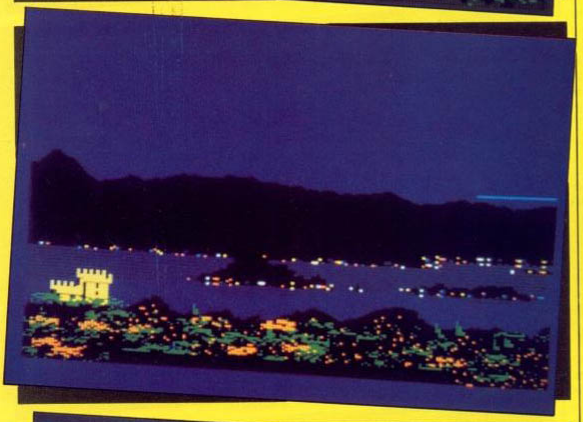
(continued on next page)



Sunrise...


...day...


...sunset...



(continued from previous page)

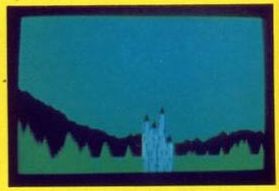
reflecting the hills and sky beyond. The precise formulation of the fundamental properties of a landscape is a subjective process — and this is where the art comes in — and the inspiration for this task came from the Highlands of Scotland and also Cumbria.

The Amstrad version of the program displays several advances compared to the Spectrum version. A tremendous advantage of Amstrad graphics is that any pixel can be any colour. You can have 16 colours in Mode 0, with a resolution of 160 pixels horizontally by 200 vertically. For better resolution, Mode 1 gives you 320 horizontally, but only four colours. I chose Mode 0 with its much greater range of colours. If you want better resolution horizontally than vertically, then you could turn the monitor on its side, and swap your x and y axes.

Pixels can be any colour

Whatever mode you select, any pixel can be any of the available colours. This gives enormous freedom compared to most home computers. Foreground features can be drawn with no effect at all on the background colours. The colours actually used on the screen can be chosen from a selection of 27.

Machine code is notoriously difficult to experiment with and so I am providing a simple program in Amstrad Basic for you to try out. It uses the RND function to generate



a recipe for a mixture of sine waves with different wavelengths and phrases. The sum of all the different waves is a complex curve which can have enormous variability. It is a well-known method called Fourier Synthesis.

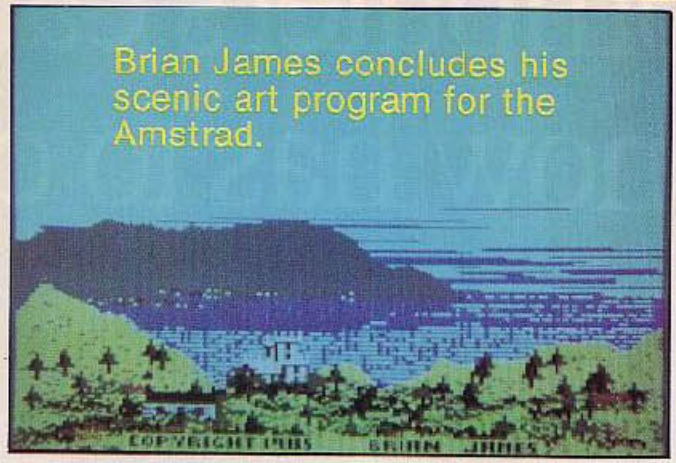
Each waveform here is drawn with a different colour and — Hey Presto! — the Fourier Series we learned in stuffy lecture theatres comes to life in brilliant colours.

The number of sine waves used in nn. You can try putting in a larger number, say nn = 20 or nn = 50. The curve becomes more and more complex — but very much slower. This method is very good for smooth curves, but becomes too cumbersome and slow for very rough crinkly lines.

Next month we will explore more possibilities with the RND function, and also peer into the fascinating world of "fractals", functions which are particularly suitable for describing the very rough mountain skylines we like to gaze upon.

I can supply the Landscape Creator doubly recorded on cassette, for the Amstrad or the Spectrum 48K. Send £5 for the Amstrad version, or £3 for the Spectrum version, to — Brian James, 21 Lamond Place, Aberdeen AB2 3UT. ■

Brian James concludes his scenic art program for the Amstrad.



LANDSCAPES

THIS MONTH we look further into the nature of probability which is the key concept in these programs for the Spectrum and Amstrad computers which generate landscape scenes spontaneously. Such a program first needs some basic knowledge about the subject to be portrayed. This knowledge is described by mathematical relationships. Some of these relationships are rigid ones — for example, solid features will hide the view behind them. Other relationships are not precisely defined; for example the slope of a mountainside may have almost any gradient, though very steep slopes occur only occasionally.

The program must be allowed freedom of choice wherever possible, but must not be allowed to do unusual things too often. That is — suitable probability functions must be used.

Listing 1 is a simple Basic program to demonstrate some probability distributions. The function RND returns a value between 0 and 1, with no bias. Therefore the combination

x = a + RND * b

gives a "flat" distribution — that is, equal probability of any value between the lower limit (a) and the upper limit (a+b). Now we can combine the RND functions in different ways to produce more subtle distributions. Suppose we put

x = a + (RND + RND + RND + RND)/4 * b

This gives a hump-shaped distribution — a crude approximation to the famous "Normal" distribution. The values of x will tend to cluster around the middle with smaller probability of extreme values. Now consider

x = a * RND ↑ 2

This will cause a skewed distribution, with small values of x having much greater probability than large values. A greater skew will be obtained from the function

x = a * RND ↑ 4

and so on. It is important to realise that RND is different every time it is used — therefore

RND + RND is NOT equal to 2 * RND

Listing 2 is a very entertaining program using two-dimensional distributions to produce a multicoloured pulsating galaxy. The same distribution can be used for a cluster of leaves, daisies etc.

These programs are in Basic, which is very easy to enter and very easy to experiment with. The Landscape Creator itself has been written in pure machine code, because of the sheer number of calculations which have to be done. Each new landscape is created in about 8-20 seconds.

Suitable probability distributions can create surprisingly realistic mountain skylines. The slope must take values which are not too steep too often, not too straight, and not showing a regular pattern. There are big bumps and small bumps and even smaller bumps . . . The mathematical functions called fractals have this property of having an endless succession of smaller and smaller wiggles, in fact going on and on, to the infinitely small.

They have been used to create mountain scenery with remarkable realism by Lucasfilm, using very expensive high-resolution equipment. However, true fractals are purely mathematical abstractions . . . they have an infinite length, because of the infinite number of wiggles on an ever-decreasing scale of size, and therefore cannot actually be shown on any video screen! However my own functions are not unrelated to fractals, since they use two different magnitudes of wiggliness.

There is a problem with the random number generators in the home computer. They are not truly random; they are "pseudo-random". This means each number is not actually independent. You will eventually come back to the same sequence and go round in a big cycle. Also patterns can sometimes be found if you plot a random scatter of points.

What should we call this kind of art? I suggest the terms "artistic algorithms" or "computer creativity". The simplest kind of program, which draws diagonal lines at random polygons, could be said to be "random art". ■

Listing 1.

```
3 ' SIMPLE DISTRIBUTIONS
4 '
5 'Brian James   April 1985
6 '
7 'AMSTRAD BASIC
8 '
9 CLS
10 KEY 7,"list "
11 n=300
12 LOCATE 1,4
13 PRINT"Flat distribution"
14 FOR i= 1 TO n
15 x=RND*640
16 PLOT x,300
17 DRAW x,310
18 NEXT
19 '
20 LOCATE 1,10
21 PRINT"Hump-shaped"
22 FOR i= 1 TO n
23 x=(RND+RND+RND+RND)*160
24 PLOT x,200
25 DRAW x,210
26 NEXT
27 '
28 LOCATE 1,16
29 PRINT"Skewed distribution"
30 FOR i= 1 TO n
31 x=RND↑8*640
32 PLOT x,100
33 DRAW x,110
34 NEXT
35 GOTO 9
```

Listing 2.

```
3 ' "PULSATING PSYCHEDELIC GALAXY"
4 '    (BEST SEEN IN DARK ROOM)
5 '
6 ' DEMONSTRATES TWO-DIMENSIONAL
7 ' HUMP DISTRIBUTIONS
8 '
9 ' Brian James   April 1985
10 '
11 'AMSTRAD BASIC
12 MODE 1:INK 0,0
13 t$="Galaxy"
14 KEY 7,"LIST "
15 KEY 5,"WHILE 1:SAVE T$:WEND"
16 ON BREAK GOSUB 33
17 '
18 FOR i=1 TO 500
19 FOR c= 1 TO 3
20 x=c*30*(RND-RND) +320
21 y=c*30*(RND-RND) +200
22 PLOT x,y,c
23 NEXT
24 NEXT
25 '
26 FOR j=1 TO 3
27 FOR i=1 TO 50:NEXT
28 IF RND<0.6 THEN c=RND*25
29 INK j,c
30 NEXT
31 GOTO 26
32 '
33 INK 1,24:END
```