COLOUR BY NUMBERS

This instalment of our series of entertaining program ideas looks at a computer game based on the old party and schoolroom favourite 'Simon Says'. Like its electronic toy counterparts, the program uses colour and sound to provide a challenging memory test.

'Simon Says' is one of the first games many of us can remember. The leader gives instructions such as 'Simon says put your hands on your head' or 'Simon says stand up', and so on. Players are disqualified if they respond to an instruction that doesn't start with 'Simon says...'

Surprisingly, this party or classroom game was turned into an electronic game when a number of dedicated electronic toys, using microprocessors to control a set of buttons, lights and buzzers, were developed. In the electronic version, the computer is the leader and every instruction must be followed.

The toy provides a sequence of tones and lights and the player or players must repeat this by pressing the appropriate keys. The toy then adds another note onto the sequence and starts again. These toys became so popular that people often think of 'Simon' games as purely electronic toys, forgetting their simple origins.

Of course, if you own a home computer, the idea of a dedicated computer toy must seem quite strange. A toy may be very portable, durable and easy to use. But even the best games soon become boring and the computer's ability to run hundreds of different programs ensures that it will never be uninteresting!

The program listed here is called 'Follow That!' and plays a game based on four coloured lights, each with its own sound and key on the keyboard (they are numbered from 1 to 4). The program illuminates one light and then asks you to match it. If you get it right, the program displays two lights, and so on.

There are two ways you can be 'caught out' at this game — you may take too long to respond with the next button or you may hit the wrong button three times in any one game. To make the game more difficult, this version also becomes faster as it goes along, although there is no need for you to repeat the sequence at the speed it was played. The game has a maximum of 50 lights in any one sequence — you are unlikely to reach this limit; most players manage around 15 lights before it all becomes too much for them!

It's worth looking at the way the program is structured. All the colour and sound commands are grouped into subroutines at the end of the program as follows:

```
1000 Display light number a
1500 Get a keypress 1,2,3 or 4
2000 Make a noise for the end of the game
2500 Make a noise to warn that a response was
     wrong
6000 Print a message on line 20 of the screen and
     pause after it if necessary
```

These subroutines might correspond to actual hardware devices on a dedicated toy. Extracting them from the main body of the program has two beneficial effects. Firstly, all the machine-specific complexities of sound and colour are kept in one place, making it easier to move the program from machine to machine. Secondly, it would be easy to use the subroutines for a different game along roughly the same lines. The same program could offer several variations on the game just as a dedicated toy does. You might like to try inventing a few extras of your own and adding these to the program. For example, the program could be adapted to allow individual scores to be displayed for any number of players.

As a computer owner, you should never forget that anything a dedicated toy can do, your computer can do better.

KEVIN JONES