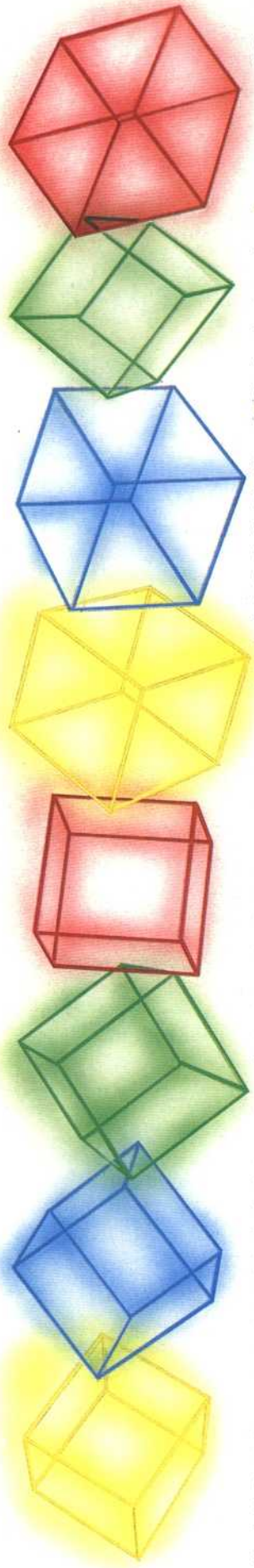
# CUBIST REVOLUTION



**Our programming projects series continues with a look at a program that allows us to draw three-dimensional shapes and rotate them. The geometrical principles used in the program to plot the points on the outlines provide a basis on which simple animated graphics can be developed.**

The program we provide uses some basic geometric principles to create perspective projections of objects that can be viewed from any angle or distance. All the data for the objects are stored in DATA statements in the program. These comprise three-dimensional co-ordinates for each end point of a line in the object. For each point there is also a figure to indicate whether it is at the beginning of a line or at its end, which tells us whether the line is to be drawn from, or to, that point.

To turn these three-dimensional co-ordinates into two-dimensional values representing points on the screen is a matter of simple, but lengthy, mathematics. The x and y co-ordinates of each point (in the plane of the television screen) are divided by a factor representing the distance of the object from us. In addition, the resulting co-ordinate is scaled by a factor suitable for the co-ordinate system of the micro. By changing the distance factor, the object can be made to shrink or grow as it approaches or recedes from the viewer.

A third constant can be used to change the effect of the perspective projection. By increasing the value of this constant the perspective view of the object is exaggerated, as though it were being viewed with a wide angle lens. Decreasing this constant gives the effect of flattening the view, as though we were looking through a telephoto lens.

In addition to creating a perspective view of the object, the program also allows it to be rotated so that it can be viewed from any angle. This is done by simple trigonometry. The axes are rotated through the desired angle so that when the perspective projection is made the image on the screen appears to have been rotated. This can be done either as a rotation around the y axis (the viewpoint appears to move around the object) or around the x axis (the viewpoint rises above or below the object).

Although our program can create all of these effects, it is remarkably simple in operation. Control of the viewpoint is through number keys one to eight. These will give, respectively: a movement of the viewpoint to the left, the right, up, down, towards the object or away from it; an increase in the perspective effect (fisheye); or a decrease in the perspective effect (telephoto).

The current three-dimensional co-ordinates are stored as three arrays: X, Y, and Z. The changes to these arrays and the changes to the constants used in the perspective transformation are made in a series of subroutines. Each time a keypress is detected, the image on the screen is erased by drawing it in the background colour, the desired change is made with a call to the right subroutine, and the new image is drawn again.

The perspective transformation is made in the subroutine that draws the image. This goes through each set of three-dimensional co-ordinates, translates them to two-dimensional co-ordinates and plots them on the screen (either moving to the points or drawing lines according to the fourth piece of data for each point).

Creating your own object data for this program is a fairly lengthy business, but very straightforward. The data is stored in statements at the end of the program, with four values for each point in the object. The total number of points is set in the first line of the program. This should be changed to suit your data. The data we give creates a cube with a diagonal line across one face.

Each set of four values is in the order: plot or draw value, x co-ordinate, y co-ordinate, z co-ordinate. The values are calculated by mentally tracing the outline of the object in three dimensions. Using an imaginary pen, visit each corner point of the object's outline in turn. If your pen is moved to a point without drawing a line, 4 is used for the first value. A value of 5 indicates that a line is to be drawn to the point from the preceding one. These particular values are used because they make the program for the BBC Micro less complicated.

The co-ordinate origin (0,0) is at the centre of the screen. It is best to make this the centre of your object. The x axis is the horizontal axis, with positive values going upwards. The z axis is the axis in and out of the screen. The positive direction on this axis is into the screen.

Keep the X, Y, and Z values as small as is reasonably possible. The initial setting of the perspective effect and viewpoint distance must take into account that an object width of about 10 will fill the screen. This could be changed by altering the scaling factor used in the perspective transformation. We suggest that you experiment with simple shapes first. Keep the number of points low. When you have mastered the digitising of simple solid objects (a pyramid, a cube) you can move on to more complex ones.

Our program could be extended further to give additional effects. A translation facility could be