

*Continued from previous page*

some random fourth place on the paper. Randomly choose one of the three dots, aim at it, go halfway towards it and make a dot. Randomly pick one of the three original points again, aim at it and move halfway towards it. Keep doing this. The pattern that emerges from all this randomness is quite striking and always the same!

After watching the program, I (and I'm sure many other Logo people) wrote a Logo program to play the Chaos Game. Here are some changes you might want to play around with:

- Instead of having the three points form an equilateral triangle, try different arrangements.
- Try four or more points instead of three.
- Instead of moving halfway towards each vertex, try different distances.
- Compare the pattern my chaos program generates to what you get with this procedure (try Hmmm 50):

```
to Hmmm :side
  if :side < 1 [stop]
  repeat 3 [forward .5 * :side
            Hmmm .5 * :side
            forward .5 * :side
            right 120]
end
```

end

(Things that make you go Hmmm. . .) Can you write other programs that always produce the same result, but via a process that is partly random?

## Find.Turtle

This is a game which helps students learn how to locate points on a cartesian coordinate plane in a problem solving mode. A turtle is hiding somewhere on the grid. The goal is to find the turtle's hiding place. The player starts by guessing the location and gets feedback from the computer. Example: TRY 3 5 might get the computer to say: GO TO LEFT and UP.

**Factor Game** *This game appeared as a CLIME Microworld in Volume II for Apple II computers. By popular demand, it is now available for the Macintosh.*

The object of this game is to get the highest score by choosing numbers from a list of counting number (i.e. 1 to 25). Player A gets the value of the number he or she chooses, while player B gets the sum of all the *factors* of Player A's number that are still on the screen. For example, if A chose 25, B gets 5 and 1 which makes the score 25 to 6. The chosen numbers are then removed from the screen and player B gets to choose. Play continues until all numbers are removed from the screen. Highest score wins. *Challenge question:* Can player A develop a strategy that will always be a winning one - assuming player A goes first?

**Histograms** *by Michael Tempel, Logo Foundation, New York, NY. This histogram program provides a visual representation of numeric data. It is based on one originally written by Brian Silverman.*

As an example, let's write a Logo program that rolls dice. Then we can use the results for the histogram. We can start with one die:

```
to die
  output 1 + random 6
end
repeat 15 [print die]
```

will display 15 "rolls" of this die. To graph this data we first set up a histogram of the appropriate size using the procedure SETUP.

## SETUP 6

draws a vertical line at the left side of the screen with the number 0 at the bottom and 6 at the top. The procedure PLOT plots a single point.

```
REPEAT 10 [PLOT 6]
```

draws a horizontal bar 10 steps long at the 6 position.

*Continued on next page*