JULIA

PURPOSE

Determine whether a series of one or more points in the complex plane belong to a given Julia set.

DESCRIPTION

Julia sets are derived from the following quadratic function:

$$Q_c(z) = z^2 + c \qquad \textbf{(EQ 6-93)}$$

where both z and c are complex numbers (z = x + iy and c = c1 + c2i). For a given point z, this function reduces to:

$$Q_c(z) = (x^2 - y^2 + c_1) + (2xy + c_2)i \qquad \textbf{(EQ 6-94)}$$

For a given point z, this function is iterated repeatedly (i.e., $Q_c(z)$ is calculated and the new complex point is feed into Qc(z) again). For a given value of c and an initial starting point, this iteration either escapes to infinity (in which case the point is said to belong to the escape set of c) or becomes bounded (the point belongs to the prisoner set). The Julia set is defined to be the points on the boundary of the escape and prisoner sets.

DATAPLOT returns the number of iterations it takes the point to escape to infinity (defined as value with an absolute value > 10**6). DATAPLOT terminates the iterations at 100 if the point has not escaped. A plot can be made of the Julia set from these counts (i.e., plot the point if the count is less than 100). This plots the prisoner set (the Julia set is the boundary of this plot).

SYNTAX

LET <count> = JULIA(<x0>,<y0>,<c1>,<c2>) <SUBSET/EXCEPT/FOR qualification>

where <count> is a parameter or variable where the number of iterations calculated for the given point (terminates at 100) is saved;

<x0> is a parameter or variable which contains the real component of the point;

<y0> is a parameter or variable which contains the imaginary component of the point;

<c1> is a parameter or variable which contains the real component of the c parameter;

<c2> is a parameter or variable which contains the imaginary component of the c parameter;

and where the <SUBSET/EXCEPT/FOR qualification> is optional.

EXAMPLES

LET Y1 = JULIA (Z1,Z2,C1,C2)

NOTE 1

The program example shows how to generate a plot for a given Julia set. Different sets can be plotted by entering different values for the c parameter (i.e., change CREAL and CCOMPLEX in the program). Also, you can zoom-in on the plot by generating the plot over a restricted range (and using a finer increment between successive points).

NOTE 2

Julia plots can be generated with nonquadratic functions. However, the current implementation in DATAPLOT is limited to quadratic functions. Also, there are alternate algorithms that plot the border points (i.,e., the actual Julia set) rather than the prisoner set. These are not implemented in DATAPLOT at this time.

NOTE 3

The appearance of the Julia plot can be controlled by the settings for the CHARACTER command. For example, you can set the size and specify whether the character is filled. Also, you can change the characteristics based on the count. The mechanism for doing this is demonstrated in the program example (the creation of the TAG variable). The most common option is to plot points with the same tag value in the same color.

DEFAULT

None

SYNONYMS

None

RELATED COMMANDS

CANTOR NUMBERS = Generate a sequence of Cantor numbers.

FRACTAL PLOT	=	Generate a fractal plot.
PLOT	=	Generates a data or function plot.

REFERENCE

"Chaos, Fractals, and Dynamics," Robert Devanney, Addison-Wesley, 1990 (chapters 5 and 6).

APPLICATIONS

Chaos, fractals

IMPLEMENTATION DATE

93/7 (earlier versions have a bug in the algorithm)

PROGRAM

```
. NOTE: FOR GREATER RESOLUTION, SPLIT THE PLOT INTO 4 PORTIONS.
LET CREAL = -1; LET CCOMPLEX = 0
LET XFIRST = DATA -2 0 -2 0; LET YFIRST = DATA 0 0 -2 -2
ERASE; PRE-ERASE OFF; XLIMITS -2 2; YLIMITS -2 2
LET INC = .02
LET C1 = CREAL FOR I = 1 1 1000; LET C2 = CCOMPLEX FOR I = 1 1 10000
CHARACTER CIRCLE BL BL BL BL BL BL BL BL; CHARACTER SIZE 0.1
LINE BLANK ALL
X1LABEL REAL COMPONENT; Y1LABEL IMAGINARY COMPONENT; TITLE JULIA SET OF Z**2 - 1
LOOP FOR K = 1 1 4
    LET TEMP1 = XFIRST(K); LET TEMP2 = TEMP1 + 2 - INC
    LET X0 = SEQUENCE TEMP1 INC TEMP2 FOR I = 1 1 10000
    LET TEMP1 = YFIRST(K); LET TEMP2 = TEMP1 + 2 - INC
    LET Y0 = SEQUENCE TEMP1 100 INC TEMP2
    LET COUNT = JULIA(X0,Y0,C1,C2)
    LET TAG = INT(COUNT/10) + 1; LET TAG = 0 SUBSET TAG >= 11
    PLOT Y0 X0 TAG SUBSET TAG = 0
END OF LOOP
```