The Mandelbrot set is intriguing for its combination of simplicity and complexity. It is based on repetitions of the Julia mapping.

The Julia Mapping

Here is the coordinate mapping that repels points from Julia fractals:

$$x' = x^2 - y^2 + a \qquad\qquad y' = 2xy + b$$

If you consider this mapping in the complex plane, you can use geometric operations to perform it. Let points (a, b), (x, y) and (x', y') represent numbers in the complex plane:

$$c = a + b\text{i}$$

$$z = x + y\text{i}$$

$$z' = x' + y'\text{i}$$

Q1 Show that the Julia mapping above can be expressed as $z' = z^2 + c$.

You'll perform this mapping using vector operations. To use vector operations on complex numbers, think of a complex number as corresponding to a vector from the origin to the point representing the complex number. The length of this vector is called the *modulus* of the complex number, and the angle it makes with the positive real axis is called the *argument*.

To *multiply* two complex numbers represented in this form, you must multiply their moduli and add their arguments. How would you *square* a complex number?

Q2 Fill in the blanks: To square a complex number, you must ____________ its modulus and _____________ its argument.

Q3 For each complex-number operation below, identify the transformation (translation, rotation, dilation, or reflection) that you would use to perform the operation geometrically:

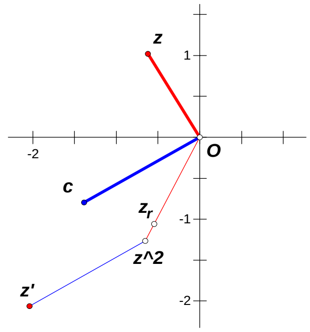
double the argument of z _____________________

square the modulus of z _____________________

add the complex number c _____________________

1. Open the Mapping page of the file **Mandelbrot.gsp** in the folder **Supplemental Activities | Mandelbrot**. Vectors z and c are shown on the complex plane. Press the *Map* button to see an animated iteration of the Julia mapping.

Press the *Reset* button on the sketch and duplicate the mapping using the Sketchpad Transform menu commands, as follows:



2. Mark angle *BOz*. Rotate point *z* about point *O* by the marked angle. Label the image point z_r.

Be sure to choose **Coordinate Distance**, not **Distance**. The measurement must be based on the coordinate grid.

3. Measure the coordinate distance *Oz*. Mark this measurement as the scale factor. Dilate point z_r by the marked ratio. Label the image *z^2*.

4. Mark vector *Oc*. Translate point *z^2* by the marked vector. Label the image *z′*. Press the *Map* button again to confirm that the mappings agree.

In the next several steps, you will use iteration to construct a Julia set.

5. Press the *Reset* button so that only your construction remains showing. Press the *Case 2* button to change the scale of the coordinate system and the initial values of *z* and *c*.

6. Hide the intermediate points (z_r and *z^2*) and the coordinate distance measurement. Also hide the segments representing the various vectors, but leave points *z*, *c*, and *z′* visible.

7. Iterate point *z* to *z′* by selecting point *z* and choosing **Transform | Iterate**. When the dialog box appears, click on point *z′* in the sketch and then press the Iterate button in the dialog box.

To increase the number of iterations, select the iterated image, choose Edit | Properties, and use the Iteration panel to change the number of iterations.

8. When the iteration appears, choose **Display | Line Width | Dashed**, and then increase the number of iterations to 200.

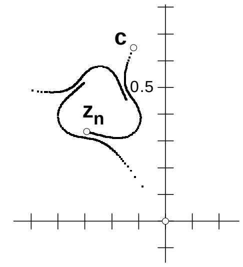
9. Experiment with different positions of *z* and *c* to view different Julia fractals. (Only certain positions of *z* and *c* make interesting fractals; for many positions, the iterated points converge quickly to a single point, or quickly fly off-screen.) You may want to print some of the more beautiful ones you find.

Testing For Inclusion

For some initial values of *z* and *c*, the iterated points fly off-screen quickly, and for other initial values the iterated points remain bounded. The Mandelbrot set is the set of points *c* for which the Julia set starting with $z = c$ is bounded.

Thus the Julia iteration does not actually draw the Mandelbrot set. Rather, you use the Julia iteration as a test to see if the starting point *c* is in the set. After many iterations of the mapping, the orbit of the iterated point may tend to converge on a single point, oscillate between two or more points, or fly off into the distance. If the orbit does not fly off, the initial point is said to be in the Mandelbrot set.

10. Go to the Iteration page. This sketch contains a Julia set iteration in which the starting value is $z = c$. Parameter n controls the number of iterations. Point z_n is the image after n iterations. Drag point c to observe the Julia set for different values of c.



It turns out that if any point in the orbit has a modulus greater than 2, the modulus must keep increasing, so the point is not in the set. This provides a simple test for exclusion. (Unfortunately, there is no simple test for inclusion. A particular orbit may contain many points with modulus less than 2 before diverging.)

11. Go back to the Mapping page. Run the mapping and press the *Show Circle* button. Move point z outside of the circle. Make the modulus of c less than or equal to that of z. Confirm that the modulus of z' is greater than that of z.

Q4 Prove that if $|z| > 2$ and $|z| \geq |c|$, then $|z'| > |z|$.

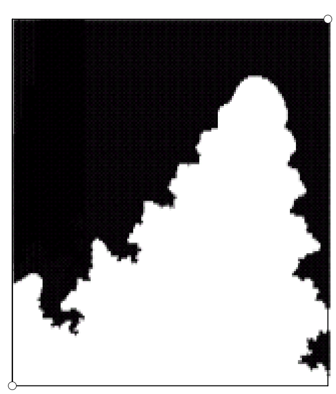
12. Go to the Mandelbrot page and press the *Paint* button. The painting process can take a while, so be patient.

As the image is painted, some points appear shaded and some are totally black. Shaded points are those whose orbits diverge and are not in the set. The lightest-colored points are those that take the greatest number of iterations to diverge. The black *continent* in the middle depicts the set of points whose orbits do not diverge.

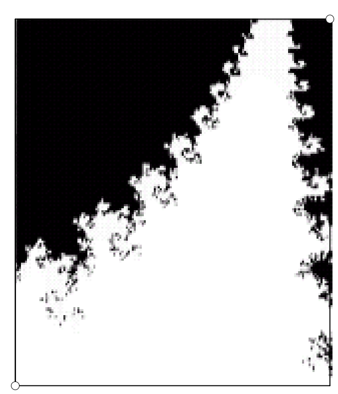
The fact that a point is black does not prove that it is in the Mandelbrot set. This only means that it does not diverge in the number of iterations indicated by the parameter *depth*. Points that are near the boundary of the set may require billions of iterations before they are persuaded to leave.

13. Drag the *zoom* window to a boundary region for closer inspection. Press *Zoom*, then *Paint*.

14. After the region has been painted, increase *depth* to 50 and paint it again.



depth = 20



depth = 50

Q5 Describe the difference in the renderings that appeared at *depth* = 20 and *depth* = 50.

THE MANDELBROT SET

Objective: Students investigate the Julia mapping and see how it is used to generate the Mandelbrot set. The accompanying sketch produces a beautiful colored image of the Mandelbrot set and allows the student to zoom in on any portion of the set.

Prerequisites: Students should be familiar with complex numbers and their representation on the complex plane. This activity is not appropriate as an introduction to these concepts.

Sketchpad Proficiency: Intermediate. You could present this activity to beginners if you skip steps 2–9.

Class Time: 40 minutes

Required Sketch: Mandelbrot.gsp

This activity is related to and follows the Chaos activity, but it can also stand alone.

The rendering on the Mandelbrot page can be slow, depending on the computer being used. You may want to show the set by using a fractal rendering program designed specifically for that task.

THE JULIA MAPPING

Q1 $z^2 + c = (x + y\text{i})^2 + a + b\text{i}$

$= x^2 - y^2 + 2xy\text{i} + a + b\text{i}$

$= (x^2 - y^2 + a) + (2xy + b)\text{i}$

$= x' + y'\text{i}$

The directions remind students of a method for multiplying two complex numbers. This method is explored in detail in the activity entitled Multiplication of Complex Numbers.

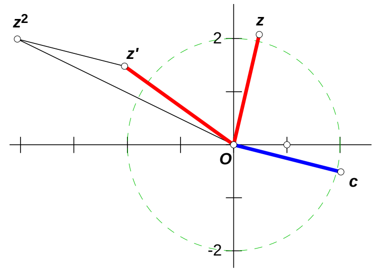
Q2 To square a complex number, you must square its modulus and double its argument.

Q3

Double the argument of z	rotation
Square the modulus of z	dilation
Add the complex number c	translation

In steps 5–9, students use their construction of a single iteration to create the first 200 iterations of various Julia fractals. In this process students gain perspective by creating their own Julia sets and seeing convergence for some values of z and c and divergence for others. But the rest of the activity does not depend on this process, so you can omit these steps if time is limited.

TEST FOR INCLUSION



Q4 Let $|z| > 2$ and $|z| \geq c$. Let O be the origin. Form a triangle with the points O, z^2, and z'. By the triangle inequality law:

$|z'| + |c| \geq |z^2|$

They are perhaps equal because the points could be collinear. Since $|z| > 2$, it follows that $|z^2| > 2|z|$.

$|z'| + |c| > 2|z|$

$|z'| > 2|z| - |c|$

$|z'| > 2|z| - |z|$

$|z'| > |z|$

Q5 The rendering with greater depth shows more detail. Also, the area of the continent decreases.

The rendering uses the modulus > 2 test to determine which points are in the set. This test will always incorrectly include some points whose orbits have not yet left, but it will not exclude any points that should be included.