

Contents

Mandelbrot/Julia Set Fractal Generator Version 6.0



[Introduction](#)

[Quick Start for Impatient New Users](#)

[Menu Commands](#)

[Image File Structure](#)

[Mathematics of the Mandelbrot Set](#)

[Selected References](#)

[Registration](#)

[Registration Form](#)

[Ombudsman Statement](#)

Introduction



Work on the earliest version of this program began in August of 1985, when the Scientific American contained an article by A. K. Dewdney in his Computer Recreations column entitled "A computer microscope zooms in for a look at the most complex object in mathematics." This article was the first exposure the general public had to Mandelbrot images and, perhaps more than many other technical articles, launched a popular interest in fractal programming. Within a few days of reading the article I had produced a black and white 50 pixel square image of the Mandelbrot set on an IBM PC with an Intel 8088 CPU, an effort which took a couple of hours of computing time as I recall.

Today I'm still hooked on these beautiful images. After about a dozen MS-DOS versions of this program, this Windows program is my latest effort to share these images with others.

Quick Start for Impatient New Users

After launching the program click on the File pulldown menu and then click on Open. Double click on the image file M1.MAN, and the complete Mandelbrot image will be displayed. This image is plotted from $-2 \leq x \leq +2$ and $-2 \leq y \leq +2$ and no other fractal features are to be found outside this range. A right mouse click will stop the display of the image.

The zoom window feature is started by clicking and dragging horizontally on the displayed image. The zoom window will now appear, with its crossed center lines. Once correctly positioned the left mouse button is released and a verify window appears. Clicking on OK will store the new parameters for the programs use.

These new parameters will be used to generate an enlarged image of the features found in the zoom window. The new parameters can be viewed and manually edited from the Edit pulldown menu by clicking on M/J Parameters. The zoom window and any new parameters can be abandoned by clicking the mouse cursor on Cancel.

Once the parameters have been changed through the use of the zoom window, you may wish to change the image file name in the M/J Parameters Window. If this is not done the original image file will be erased when the new one is generated. As every image is generated, an image file will be created using the current image file name. The image file name must use the .MAN extension.

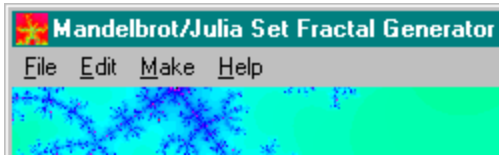
Clicking on the Make pulldown menu, and then M/J Image, will start the generation of the new zoomed image.

If you inadvertently erase the original M1.MAN image's initial parameters they can be easily restored by typing them into the M/J Parameter Window. They are:

X center = 0
Y center = 0
Magnification = 0.5
Maximum dwell = 511
Image width = 640
Palette number = 5

Once entered, just regenerate the image by clicking on the Make pulldown menu and then M/J Image.

Menu Commands



File (New, Open, Save As, Print and Exit)

Edit (M/J Parameters and Chaos Parameters)

Make (M/J Image and Chaos Image)

Help (Contents, Print Registration Form, Our Phone Number and About)

File Commands



New

The New command simply erases the screen and allows the user to begin a new image. The larger of the Mandelbrot/Julia or Chaos widths is used to determine the square dimension of the blank screen. If neither of these two values is greater than the screen window plus 20 pixels, the square dimension is set to this value.

Open

The Open command allows the user to display a previously stored image and load the parameters of the image. A dialog box is opened allowing the user to select a file. Three different type of images can be opened. These are files with .MAN (Mandelbrot), .CHS (Chaos) and .BMP (Windows bitmap) extensions. After a .MAN file is opened and displayed, the zoom feature will be enabled. A .CHS file will regenerate itself when opened.

Save As

The Save As command will open a dialog box allowing the user to type in a file name. This file name should have the .BMP extension, as saved pictures are stored in the Windows bitmap format. The image saved in Windows bitmap format will consist of only the visible window display.

Files with .MAN and .CHS extensions are not saved with this command. These files are automatically created as the Mandelbrot, Julia and Chaos images are generated.

Print

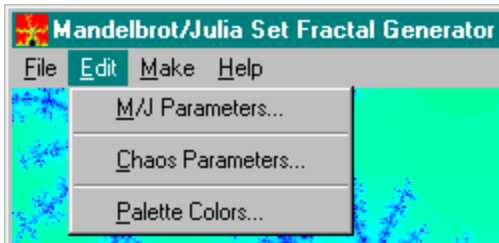
The Print command brings up a standard print dialog box. Clicking OK starts the printing process of the image currently displayed.

Be sure you have the correct print driver for your printer assigned as the default print driver in the Print Manager. The Print Manager is found in the Main window in Windows 3.1, and in the Control Panels in My Computer in Windows 95. When you click on Print, a Print Dialog box appears and clicking on the Setup button will allow you to choose settings for your printer which may enhance the appearance of your printed image.

Exit

The Exit command will quit the program after displaying an "Are you sure?" message.

Edit Commands



M/J Parameters

Clicking on this menu item allows you to type in specific initial parameters for a Mandelbrot or Julia image. Refer to the Mathematics of the Mandelbrot Set section for details about the individual parameters. When a Mandelbrot or Julia image is generated, the program automatically saves a .MAN file which contains all the initial parameters and the dwell values computed for every point.

Chaos Parameters

Clicking on this menu item allow you to type in specific initial parameters for a Symmetrical Chaos image. Refer to "Symmetry in Chaos, A Search for Pattern in Mathematics, Art and Nature", by Michael Field and Martin Golubitsky for more details on these parameters. Changes in these values can be very sensitive so if you change them make very small changes and check the effect. When a Chaos image is generated, the program automatically saves a .CHS file which contains the initial parameters, but no other information about the image. When a .CHS image is opened from the File menu, the image is regenerated.

The following are some sample values to use for chaos images.

Sand Dollar

lambda = -2.34
alpha = 2.0
beta = 0.2
gamma = 0.1
omega = 0
n = 5

Mayan Bracelet

lambda = -2.08
alpha = 1.0
beta = -0.1
gamma = 0.167

$\omega = 0$
 $n = 7$

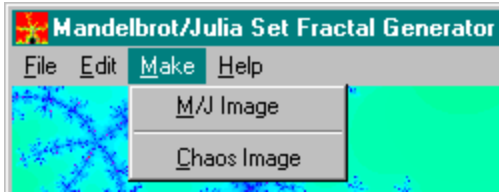
Swirling Streamers

$\lambda = -1.86$
 $\alpha = 2$
 $\beta = 0$
 $\gamma = 1.0$
 $\omega = 0.1$
 $n = 4$

Palette Colors

This menu item brings up a small window with a 256 color bar which represents the current palette. Clicking on Next or Back will change the palette used for coloring images. Clicking on OK selects the displayed palette, clicking on Cancel makes no change to the palette. There are 12 different palettes available.

Make Commands



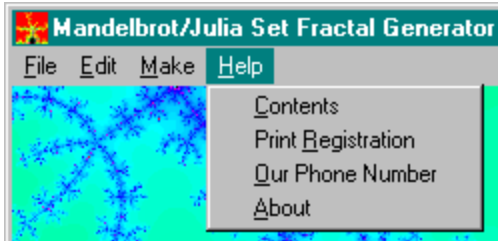
M/J Image

When you click on this menu item a Mandelbrot or Julia image is generated using the current M/J parameters. Generation can be stopped by clicking on the image with the right mouse button. After the image has been generated the zoom feature is enabled.

Chaos Image

Selecting this menu item will begin the generation of a new Symmetrical Chaos image using the current chaos parameters. Generation will continue for a total of 300,000,000 points unless manually stopped by clicking on the image with the right mouse button. Depending on the palette used and the desired depth of colors, the user should continue to monitor the progress of the image and stop generation when the desired effect is reached.

Help Commands



Contents

Selecting Contents brings up this Help Window.

Print Registration Form

Clicking on this will print out a copy of the registration form. When you register (pay for) this software you received a copy of the latest version of this program along with additional images.

Our Phone Number

Clicking this pops up our Customer and Technical Support phone number, 847-854-0489, for easy reference and orders.

About

The standard About window, who and where we are.

Image File Structure

Mandelbrot and Julia Images

Each image file created by the Mandelbrot/Julia Set Generator begins with a 128 byte header.

Byte	Item	Size	Description
0	x	8 byte double	x center point
8	y	8 byte double	y center point
16	mag	8 byte double	magnification
24	a	8 byte double	a for Julia sets
32	b	8 byte double	b for Julia sets
40	maxdwell	unsigned int	maximum dwell
42	width	unsigned int	image width in pixels
44	mj[2]	char	M/J, image type
46	palette	int	palette number 0-11
48	fill[30]	char	filler
78	name[50]	char	Wierenga Software signature

All char strings are terminated with a hexadecimal 00 byte.

The dwell data follows the header. It should be noted that this is not a true image file, rather the dwell values themselves are stored. This allows users to color the image with different palettes. Storing an image file might be simpler but for every different coloring of the image a new image file would have to be created.

The dwell data is stored as a series of two byte unsigned integers. Each unsigned integer contains the dwell value and a run length corresponding to a string of identical dwell values. The number of bits required to hold the maximum dwell is first obtained. If the maximum dwell is 511, then 9 bits are required, 1023 would require 10 bits, etc. Using 1023 for the maximum dwell as an example, the right most 10 bits of the 16 bit integer represents the dwell value and the 6 left most bits contain the run length. As a run length of zero is not very useful, this value is always incremented by one such that a run length of zero equals 1, 1 is 2, etc. Given a maximum dwell of 1023 the following 16 bit unsigned integer represents a dwell of 1000 and a run length of 32.

```
011111 1111101000
       7FE8 hex
```

When an image is being displayed and the unsigned integer above is read, a line of 32 pixels will be drawn using the appropriate color from the active palette for dwell value 1000, ie. $\text{palette number} = \text{mod}(1000/256)$.

Each line of a display is encoded with no wraparound. This means that each line will end

with the display of an encoded unsigned integer and no extra pixels of the same dwell will be added for the beginning of the next line even if there is room in the run length.

It should be noted that the maximum run length that can be stored varies with the maximum dwell chosen. Files with a maximum dwell of 1023 will have a maximum run length of 64, those with maximum dwells of 8191 will only store 16. This does not limit a run length because if it exceeds the space available in a single unsigned integer it simply creates additional ones until the run of dwells has been stored. For this reason images with high maximum dwell values are often large in size. This method of file compression strikes a good balance between file size and speed when displaying an image.

Note that choosing the maximum dwell value to be in the set $2^n - 1$ maximizes the file compression method the Mandelbrot/Julia Set Generator program uses. In addition, the last entry in the palette is black, which insures that images with the maximum dwell set to $2^n - 1$ will color points in the Mandelbrot set black.

Symmetrical Chaos Images

Each chaos file created by the Mandelbrot/Julia Set Generator consists of the following 98 bytes.

Byte	Item	Size	Description
0	lambda	8 byte double	lambda value
8	alpha	8 byte double	alpha value
16	beta	8 byte double	beta value
24	gamma	8 byte double	gamma value
32	omega	8 byte double	omega value
40	n	signed integer	symmetry
42	width	signed integer	image width in pixels
44	plot	signed integer	plot colors linear/exponential
46	palette	signed integer	palette number 0-11
48	name[50]	char	Wierenga Software signature

No additional data beyond these initial parameters is stored in a .CHS file.

Mathematics of the Mandelbrot Set

The Mandelbrot set is computed by operating on a fairly simple equation that contains complex numbers of the form

$$x + yi \text{ where } i = \sqrt{-1}$$

The Mandelbrot equation is

$$z \leftarrow z^2 + c$$

where

$$z = x + yi \text{ and } c = a + bi$$

substituting these values into $z^2 + c$ we have

$$(x + yi)^2 + a + bi$$

$$x^2 + 2xyi - y^2 + a + bi$$

separating the real and imaginary parts of z gives

$$x \leftarrow x^2 - y^2 + a$$

$$y \leftarrow 2xy + b$$

To determine whether a point (a,b) in the complex plane is a member of the Mandelbrot set, the real and imaginary parts of the equation are iterated. The x and y values are first initialized to zero. The constants a and b , the point in the plane, are then substituted into the equations giving

$$x \leftarrow a \text{ and } y \leftarrow b$$

for the first iteration.

The two new values for x and y , along with the constants a and b , are now substituted into the equations again. This procedure (iteration) continues until the absolute value of $x + yi > 2$, ie. $\sqrt{x^2 + y^2} > 2$. For those cases where this value never exceeds 2, the maximum number of iterations is preset. A value of about 500 is usually adequate, although this value is raised to several thousand when smaller details at high magnification are examined. The number of times the equations are iterated before the value of $\sqrt{x^2 + y^2} > 2$ is called the dwell. Those initial points (a,b) where the dwell is infinite, or for more practical purposes attains the preset maximum, are members of the Mandelbrot set. Another way to describe this is to say that for points within the Mandelbrot set, the sequence of points produced by this iteration procedure is bounded inside a circle of radius 2, where points outside the set are unbounded and continue to grow and escape the circle.

The Mandelbrot set exists entirely within the area defined by

$$-2 \leq a \leq 2 \quad \text{and} \quad -2 \leq b \leq 2$$

in the complex plane. A Mandelbrot image is produced by taking this area of the complex plane and dividing it into a array of 1200 x 1200 points. Each one of these points becomes the constant (a,b). The iteration procedure previously described is used on each of the 1.44 million points, coloring each point in the Mandelbrot set black and all others white. The algorithm is:

```
maxcount <- 1000
for b <- -2 to 2 stepdown 1/300
for a <- -2 to 2 step 1/300
  x <- 0
  y <- 0
  count <- 0
  while sqrt(x^2 + y^2) < 2 and count < maxcount
    x <- x^2 - y^2 + a
    y <- 2*x*y + b
    count <- count + 1
  end while
  if count = maxcount plot(a,b,BLACK)
  else plot(a,b,WHITE)
end for a
end for b
```

While the algorithm is not that complex, the amount of computation is enormous. Depending on programming language and style, the inner loop has at least four multiplications and a square root. For a point in the Mandelbrot set this loop is executed 1000 times and there are over a million points to check! It is not surprising that the Mandelbrot set was not discovered until the age of computers.

In the Mandelbrot/Julia Set Generator program some additional refinements are made to standardize the initial parameters used to generate a specific image. Instead of defining the range of (a,b) values used for an area, a center point and a magnification are specified. The center point is simply a chosen (a,b) value. The length of a side which encloses the area of interest is defined as

$$\text{side} = 2/\text{magnification}$$

The following values can now be defined

$$a_minimum = a_center - \text{side}/2$$

$$b_maximum = b_center + \text{side}/2$$

$$\text{gap} = \text{side}/\text{width}$$

where width is defined as the number of points that make up a side (or on a computer screen the number of pixels), and the gap being the distance between each point.

The Mandelbrot set is an interesting image, a sort of cardioid with a spiked head attached at the left. The boundary of the set sprouts self similar buds of different sizes. Vastly more interesting images are forthcoming when we examine the boundary of the Mandelbrot set under higher magnification. To obtain higher magnifications we simply divide a smaller area into our array of points. For example, the area defined by the center point (-0.77,0.17) and magnification 20 is located in the upper valley between the head and the cardioid shaped body.

If we continue with these magnifications, very different and interesting images can be produced by coloring the dwell values in specific ways. Along with coloring points in the Mandelbrot set black, we can assign different colors to other points based upon their dwell value. For example, we might assign yellow to dwell values in the range 10 to 19, red to 20 to 29, etc. When we do this a great deal more detail begins to appear in the boundary regions. This region of interest exists only in a narrow band just outside the Mandelbrot set.

This program uses a 256 color palette to color the dwell values. A dwell value of 10 is assigned color index 10 in the 256 color palette, dwell 11 is assigned index 11, etc. Dwell values over 255 are assigned a value of $\text{mod}(\text{dwell}/256)$. The color palettes contain smooth transitions of one color to another creating very attractive images.

Generating Julia set images is a similar process. The point (a,b) is chosen from one of the interesting boundary areas of the Mandelbrot set. This value is held constant and the (x,y) value is initialized to the various points in the complex plane defined by

$$-2 \leq x \leq 2 \quad \text{and} \quad -2 \leq y \leq 2$$

This would be a magnification of 0.5, actually the Julia image can often be enlarged slightly to fill the screen and magnifications from 0.6 to 0.9 are often used.

The algorithm for generating a Julia set is

```
maxcount <- 1000
a <- constant
b <- constant
for y <- -2 to 2 stepdown 1/300
for x <- -2 to 2 step 1/300
  count <- 0
  while sqrt(x^2 + y^2) < 2 and count < maxcount
    x <- x^2 - y^2 + a
    y <- 2*x*y + b
    count <- count + 1
  end while
  if count = maxcount plot(a,b,BLACK)
  else plot(a,b,WHITE)
```

```
end for x  
end for y
```


Selected References

- Barnsley, Michael, *Fractals Everywhere*. San Diego, CA: Academic Press, 1988.
- Briggs, John and Peat, F. David *Turbulent Mirror*. New York: Harper & Row, 1989.
- Devaney, Robert L. *Chaos, Fractals, and Dynamics*. Menlo Park, CA: Addison-Wesley, 1990.
- Devaney, Robert L. and Keen, Linda, Editors. *Chaos and Fractals, The Mathematics Behind the Computer Graphics: Proceedings of Symposia in Applied Mathematics*. Providence, RI: American Mathematical Society, 1989.
- Field, Michael and Golubitsky, Martin. *Symmetry in Chaos, A Search for Pattern in Mathematics, Art and Nature*. New York, NY: Oxford University Press, 1992.
- Gleick, James *Chaos, Making a New Science*. New York: Viking Penguin, Inc., 1987.
- Mandelbrot, Benoit B. *The Fractal Geometry of Nature*. New York: W. H. Freeman and Co., 1983.
- Peitgen, Heinz-Otto and Richter, Peter H. *The Beauty of Fractals, Images of Complex Dynamical Systems*. Berlin: Springer-Verlag, 1986.
- Peitgen, Heinz-Otto and Saupe, Dietmar, Editors. *The Science of Fractal Images*. New York: Springer-Verlag, 1988.
- Pietgen, Heinz-Otto, Jurgens, Hartmut and Saupe, Dietmar *Fractals for the Classroom, (Volumes I & II)*, New York: Springer-Verlag, 1992. (There is a single volume work entitled *Chaos and Fractals, New Frontiers of Science*, which is essentially the same work as the two volume set above.)
- Pickover, Clifford A. *Computers Pattern Chaos and Beauty: Graphics from an Unseen World*. New York: St. Martin's Press, 1990.
- Pickover, Clifford A. *Computers and the Imagination: Visual Adventures Beyond the Edge*. New York: St. Martin's Press, 1991.
- Pickover, Clifford A. *Mazes for the Mind*. New York: St. Martins Press, 1992.
- Schroeder, Manfred *Fractals, Chaos, Power Laws, Minutes from an Infinite Paradise*. New York: W.H. Freeman and Co., 1991.
- Stevens, Roger T. *Fractal Programing in C*. Redwood City, CA: M&T Publishing, Inc., 1989.
- Stevens, Roger T. *Advanced Fractal Programing in C*. Redwood City, CA: M&T Publishing,

Inc., 1990.

Stewart, Ian Does God Play Dice? The Mathematics of Chaos. Oxford: Basil Blackwell, 1989.

Stewart, Ian and Golubitsky, Martin Fearful Symmetry, Is God a Geometer? Oxford: Blackwell, 1992.

Registration

This above all,-- to thine own self be true;
And it must follow, as the night the day,
Thou canst not then be false to any man.
Hamlet, Act i, Sc. 3.

If you use the Mandelbrot/Julia Set Fractal Generator, please support the author and register (buy) your copy. You will receive the current version of the program on disk along with many additional backgrounds and images. You can pay for your registration in several ways.

1. Mail your check or money order (\$22. in U.S. funds drawn on a U.S. bank) to: Theron Wierenga, P.O. Box 595, Muskegon, MI 49443. The registration form can be printed by selecting "Print Registration" under the Help menu.

2. Call our Customer and Technical Support line at 847-854-0489, and pay with your Visa or MasterCard.

3. Send an email with your Visa or MasterCard number and expiration date to twiereng@remc4.k12.mi.us.

Registration Form
Mandelbrot/Julia Set Fractal Generator, Version 6.0

Name _____

Address _____

City _____ State _____ Zip _____

Email address (if available) _____

Registration fee \$20.00 _____

Registration fee (Outside the USA) 30.00 _____

Borland C/C++ program code 20.00 _____

Shipping _____
2.00 _____

Total enclosed _____

(All prices are in U.S. dollars.)

Method of payment: Check or MO _____ MasterCard _____ Visa _____

Account number _____ Expir. date _____

Signature (necessary) _____

How did you receive your copy of this program? _____

Suggested improvements _____

Mandelbrot/Julia Set Fractal Generator, Version 6.0
is a software product of

Theron Wierenga
P.O. Box 595
Muskegon, MI 49443
Customer support and order phone 847-854-0489
twiereng@remc4.k12.mi.us
<http://www.novagate.com/~wierenga/>

Ombudsman Statement

This program is produced by a member of the Association of Shareware Professionals (ASP). The ASP wants to make sure that the shareware principle works for you. If you are unable to resolve a shareware-related problem with an ASP member by contacting the member directly, ASP may be able to help. The ASP Ombudsman can help you resolve a dispute or problem with an ASP member, but does not provide technical support for members' products. Please write to the ASP Ombudsman at 545 Grover Road, Muskegon, MI 49442-9427 USA, FAX 616-788-2765 or send a CompuServe message via CompuServe Mail to ASP Ombudsman 70007,3536 or internet 70007.3536@compuserve.com.

