FractInt defaults

This tutorial is for users of FractInt who want to use Fractal Zoomer.

Most fractals from FractInt uses a default color palette. This is included with Fractal Zoomer.

Click on "Palette (out)", then "Legacy FrantInt maps", "Default".

FractInt by default uses 150 interations, bailout of just 4 and bailout condition "circle (Euclidean norm)", in FractInt this is called "mod".

How to export a color palette from FractInt

This is sometimes required for “saved command set”, as these fractals usually have their own special color palette.

Run fractal. Hit “C” followed by “E”. FractInt will ask where to save the map (“palette”) file. Remember where you have saved the palette.

How to import a color palette in FractalZoomer

Click on “palette (out)”, click on “direct palette” or simply press “insert” on your keyboard. Choose which map file to import, click on “Load direct palette”.

There are three ways running a fractal.

The first way are the builtin fractals. You can find the fractal in the menu. Click on "Options", "Fractal options", "Fractal functions" and finally the name of fractal. You can for example find the original Mandelbrot under "Mandelbrot type" and "Mandelbrot z =z ^ 2 + C".

The second way is to enter a user formula. You can do this by clicking on "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter for example next to "z = ": sin(z*z) and press "ok".

The third is to program the fractal, which allows more complex fractals. You can do this by clicking on "fractal", "fractal options", "fractal functions", "user formulas", "user formula" and finally click "edit user code". Just remember which function you have altered. The functions starting with f, can only have one parameter. G can have two parameters. m can have as much as 10 parameters. Don't forget to safe the file. Don’t forget to click on "compile user code". Run with by entering next to "z = " for example: m1(z,c, 1).

Some programming tips

The sign function is not implemented in fractal zoomer. The function "sign()" returns 1 if the argument is positive, -1 if argument is negative. However, it’s possible to program it. Add the following before or after a preprogrammed function:

```
static double sign(double x) {
if(x >= 0) return 1;
return -1;
}
```

cabs is the norm function in fractal zoomer
recip is the rec function in fractal zoomer
ident(x+iy) = x + iy so basically this function leaves the value unchanged.

for e^z, just do z.exp();
for log of base e, do z.log()
for log of base n we will have to use the property log(X) / log(n), so z.log().divide(new Complex(n, 0).log()) or z.log().divide(Math.log(n))
cotan is the cot function, so do z.cot()
cotanh is the coth function, so do z.coth()

Cosxx(z) == conj(cos(z))

Zero. Returns 0. This allows you to "turn off" an expression without rewriting the formula. If you were using a Mandelbrot mutation with the iterated section "z = z*z + c + fn1(z)", you could see the normal Mandelbrot set by setting fn1() to zero(). Zero(1,-3) == (0,0).

Hints

Change image size to a low resolution for faster calculation. When it looks interesting, you can increase the resolution. “Options”, “Image Size”.

Set Iterations to 150 or lower for faster calculation. When it looks a bit interesting, you can always increase the iterations a lot. “Options”, “Iterations”, “Set Iterations”.

A high bailout such as 32 is often pointless. Start with 2 or 4. Sometimes even a bailout of 1.2 has effect. “Options”, “Bailout”.

Enable smoothing “options”, “colors”, “processing”, “smoothing” for nicer pictures. This can take a while.

Enable Anti-Aliasing to make the lines appear nicer. This can take a while.

Have a lot of fun when zooming in! Try color cycling! Try interesting features like “Domain Coloring”.

How to save a fractal

Generally: “File”, “Save settings as”. Give a useful name to the file.

When fractal is programmed using “edit user code”, save or backup UserDefinedFunctions.java as well.

If you just want to save a pretty picture: “File”, “Save image as”.

Julia set

After entering a fractal formula, you can check the julia version of the fractal. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following example values: Real "0.6" Imaginary "1.2".

FractInt fractals

FractInt uses many fractals. Here is an attempt to replicate those old fractals.

barnsleym.

In fractal zoomer this is called "Barnsley 1" fractal. Change bail-out to just 2.

barnsleyj1.

In fractal zoomer this is called "Barnsley 1" fractal. Change bail-out to just 2. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia) values: Real "0.6" Imaginary "1.2".

barnsleym2.

In fractal zoomer this is called "Barnsley 2" fractal. Change bail-out to just 2. Rotate 180°. Options → rotation → set rotation. Enter 180 above.

barnsleyj2.

In fractal zoomer this is called "Barnsley 2" fractal. Change bail-out to just 2. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following values: Real "-0.6" Imaginary "-1.1".

barnsleym3.

In fractal zoomer this is called "Barnsley 3" In fractal zoomer this is called "Barnsley 3" fractal. Use the builtin function of fractal zoomer. Change bail-out to just 2.

barnsleyj3.

In fractal zoomer this is called "Barnsley 3" fractal. Change bail-out to just 2. Enter the following Julia seed values: Real "-0.1" Imaginary "-0.4".

cmplxmarksjul.

Start up the regular mandelbrot. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following values: Real "0.3" Imaginary "0.6".

complexnewton.

"fractal", "fractal options", "fractal functions", "root finding methods", "newton method", "newton 3".

fn(z)+fn(pix).

Set bailout to 8. Iterations to 150.

Enter the following use formula: “sin(z)+(c^2)*(1+0i)”. Where sin is the first function. ^2 is the second function. To use sin as second function use this formula: sin(z)+sin(z)*(1+0i). (1+0i) is the parameter, with 1 as “real” part.

How to program it: Enter the following user formula:"edit user code". You can change the function by changing sin to cos for example.

```
public static Complex m1(Complex z, Complex c, Complex factor, Complex z4, Complex z5, Complex z6, Complex z7,
Complex z8, Complex z9, Complex z10)
{
                return z.sin().plus(c.sin().times(factor));

}
```

How to run it:
examples: "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter next to "z = ": m1(z,c, 1) or m1(z,c, 1-2i) and press "ok".

fn(z*z).

Set bailout to 8. Iterations to 150. Enter the following user formula: Enter next to "z = ": sin(z*z) and press "ok".

Or programatically via “edit user code”:

```
public static Complex f4(Complex z) {

        return (z.times(z).sin());

}
```

Use “f4(z)” to call the function in “user formula”.

fn*fn.

Set bailout to 8. Set iterations to 150. Enter the following user formula: Enter next to "z = ": sin(z)*sin(z) and press "ok".

Or programatically via “edit user code”:

```
public static Complex f5(Complex z) {

                return z.sin().times(z.sin());

}
```

Use “f5(z)” to call the function in “user formula”.

fn*z+z

Set bailout to 8. Enter the following user formula: Enter next to "z = ": (1*(sin(z))*z)+(1*z) and press "ok". e.g. (1-

2i)*(cos(z))*z)+(1-2i*z)

Or programatically via “edit user code”:

```
public static Complex m16(Complex z, Complex f1, Complex f2, Complex z4, Complex z5, Complex z6, Complex z7, Complex z8, Complex z9, Complex z10) {

                return f1.times(z.sin().times(z)).plus(f2.times(z));

}
```

Use “m16(z,1,1)” to call the function in “user formula”.

fn+fn

Set bailout to 8. Iterations to 150. Enter the following user formula: Enter next to "z = ": 1*sin(z)+1*sin(z) and press "ok".

Or programatically via “edit user code”:

```
public static Complex m17(Complex z, Complex f1, Complex f2, Complex z4, Complex z5, Complex z6, Complex z7, Complex z8, Complex z9, Complex z10) {

 return f1.times(z.sin()).plus(f2.times(z.sin()));

}
```

Use “m17(z,1,1)” to call the function in “user formula”.

frothybasin

Use the builtin “Frothy basin” fractal of fractal zoomer.

julfn+zsqrd

Enter the following user formula: Enter next to "z = ":(sin(z))+(z^2)+c and press "ok". "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "-0.5" Imaginary "0.5".

Or programatically via “edit user code”:

```
public static Complex g2(Complex z, Complex c) {

 return z.sin().plus(z.square().plus(c));
 }
```

Use “g2(z)” to call the function in “user formula”. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "-0.5" Imaginary "0.5".

julia.

Classic Julia set fractal.

Use the builtin Mandelbrot z^2 +c. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on

keyboard. Enter the following (julia seed) values: Real "0.3" Imaginary "0.6".

Or programatically via “edit user code”:

```
public static Complex g3(Complex z, Complex c) {

return z.square().plus(c);
 }
```

Call with “g3(z,c)” next to z: in user formula.

"Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.3" Imaginary "0.6".

julia4.

In fractal zoomer this is called "Mandelbrot z ^4 + c". Click on "fractal", "fractal options", "fractal functions", "Mandelbrot type" and lastly click on ""Mandelbrot z ^4 + c". Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.6" Imaginary "0.55".

Or programatically via “edit user code”:

```
public static Complex g5(Complex z, Complex c) {

return z.pow(4).plus(c);

}
```

Call with “g5(z,c)” next to z: in user formula. Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.6" Imaginary "0.55".

julia_inverse.

Start up the regular mandelbrot. Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "-0.11" Imaginary "0.6557". Hit "options", "colors", "processing", "Fake distance estimator", check enable and enter factor "4", check "Invert colors". Hit "options", "bailout-condition", "skip bailout condition iterations" and enter 10.

julzpower.

Enter the following user formula: Enter next to "z = ": z^2+c and press "ok". 2 can be changed to whatever you want. Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.3" Imaginary "0.6".

Or programatically via “edit user code”:

```
public static Complex m18(Complex z, Complex c, Complex h, Complex z4, Complex z5, Complex z6, Complex z7,
Complex z8, Complex z9, Complex z10) {

                return z.pow(h).plus(z.square().plus(c));

}
```

Call with “m18(z,c,2)” next to z: in user formula. Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or

simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.15" Imaginary "0.3".

julzzpwr.

Enter the following user formula: Enter next to "z = ": z^z+z^2+c and press "ok". Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "-0.3" Imaginary "0.3".

Or programatically via “edit user code”:

```
public static Complex m19(Complex z, Complex c, Complex h, Complex z4, Complex z5, Complex z6, Complex z7, Complex z8, Complex z9, Complex z10) {

	return z.pow(z).plus(z.pow(h).plus(c));
}
```

Call with “m19(z,c,2)” next to z: in user formula. Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "-0.3" Imaginary "0.3".

lambda.

Classic Lambda fractal. ‘Julia’ variant of Mandellambda.

In fractal zoomer this is called "lambda". Click on "fractal", "fractal options", "fractal functions", "lambda type" en choose "lambda". Hit "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.85" Imaginary "0.6". Alternative: enter user formula: (0.85+0.6i)*z*(1-z) Try changing (0.85+0.6i) to (0.7+ 0.725i) or (0.69 + 0.725i).

Or via user formula: lambda*z*(1-z) where lambda is parameter. For example: (0.85+0.6i)*z*(1-z) where 0.85 is real part parameter and 0.6 is imaginary part of parameter.

Or programatically via “edit user code”:

```
public static Complex g5(Complex z, Complex h) {

Complex lambdaone = new Complex(1, 0);
return z.times(h).times(lambdaone.sub(z));

}
```

Call with g5(z,(0.85+0.6i))

lambda(fn||fn).

In fractal zoomer this is called "lambda". Click on "fractal", "fractal options", "fractal functions", "lambda type" en choose "lambda fn||fn".

lambdafn.

Enter user formula: (1+0.4i)*sin(z). Set bailout to 64.

Or programatically via “edit user code”:

public static Complex g6(Complex z, Complex h) {

Complex lambdaone = new Complex(1, 0);
return z.times(h).times(z.sin());

}

Call with g6(z,(1+0.4i)).

magnet1j.

In fractal zoomer this is called "magnet". Click on "fractal", "fractal options", "fractal functions", "magnet type" en choose "Magnet1". Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0" Imaginary "0".

magnet1m.

In fractal zoomer this is called "magnet1". Click on "fractal", "fractal options", "fractal functions", "magnet type" en choose "Magnet1".

magnet2m.

In fractal zoomer this is called "magnet2". Click on "fractal", "fractal options", "fractal functions", "magnet type" en choose "Magnet2".

magnet2j.

In fractal zoomer this is called "magnet2". Click on "fractal", "fractal options", "fractal functions", "magnet type" en choose "Magnet2". Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0" Imaginary "0".

mandel4.

In fractal zoomer this is called "Mandelbrot z ^4 + c". In fractal zoomer this is called "Mandelbrot z ^4 + c". Click on "fractal", "fractal options", "fractal functions", "Mandelbrot type" and lastly click on "Mandelbrot z ^4 + c".

mandelfn.

Set bailout to 64.

Use as user formula: c*sin(z).

Or programatically via “edit user code”:

public static Complex g7(Complex z, Complex c) {

return c.times(z.sin());

}

Call with g7(z,c).

mandellambda.

Use the builtin fractal “lambda” under “lambda type”.

Or use as user formula: c*z*(1-z). Set initial value to 0.5. “Options”, “Fractal Options”, “Initial Value”. Or simply “shift-I” on keyboard. Enable “initial value” by clicking on the checkbox. Set “0.5” as real value.

manfn+exp.

Use as user formula: (sin(z))+(e^z)+c. Set bailout to 2.

Or programatically via “edit user code”:

```
public static Complex g8(Complex z, Complex c) {

Complex manfne = new Complex(Math.E, 0);
return z.sin().plus(manfne.pow(z)).plus(c);

}
```

Call with g8(z,c).

manfn+zsqrd.

‘Mandelbrot-equivalent’ for the Julfn+zsqrd fractal.

Use as user formula: (sin(z))+(z^2)+c

Or programatically via “edit user code”:

```
public static Complex g9(Complex z, Complex c) {

return z.sin().plus(z.pow(2)).plus(c);

}
```

Call with g9(z,c).

manlam(fn||fn)

Set bailout to 64.
“Options”, “fractal options”, “Fractal functions”, “user formulas”, “user formula conditional”.

Change left operand to “norm(z)”
Change right operand to “10”
Set “f(z) when left > right” to second function. e.g (z^2)*c
Set “f(z) when left = right” to second function e.g. (z^2)*c
Set “ f(z) when left < right” to first function e.g. sin(z)*c

manowar.

In fractal zoomer this is called "Manowar".

manowarj.

In fractal zoomer this is called "Manowar". Click "Tools", "julia" of simply press "J". Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0" Imaginary "0".

manzzpwr.

Use as user formula: (z^z)+(z^(2+0i))+c

marksjulia.

Use as user formula: c^(exp(-1))*z^2+c Click "Tools julia" of simply press "J". Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.1" Imaginary "0.9".

phoenix.

Use the builtin Phoenix fractal.
"Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.56667" Imaginary "-0.5".

phoenixcplx.

Use the builtin Phoenix fractal.
"Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.2" Imaginary "0.3".

popcornjul.

Set iterations to 500 for more detail. Set bailout to 2.

How to program it: Enter the following user formula:"edit user code". Change the code of "gx" where x stand for 1,2,3,4 etc. In this example we use "g4". This factal takes it time to calculate.

```
public static Complex g4(Complex z, Complex c) {

Complex h = new Complex(0.05, 0);
```

```
Complex cYn = c.times(z.getIm());
Complex cXn = c.times(z.getRe());

Complex a1 = (cYn.tan().plus(z.getIm())).sin().times(h);
Complex a2 = (cXn.tan().plus(z.getRe())).sin().times(h);


double xn1 = z.getRe() - a1.getRe() - a2.getIm();
double yn1 = z.getIm() - a2.getRe() - a1.getIm();

  return new Complex(xn1, yn1);

}
```

How to run it:
examples: "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter next to "z = ": g4(z,3) and press "ok".

Spider.

In fractal zoomer this is called "spider".

Sierpinski.

Sierpinski gasket – Julia set producing a “Swiss cheese triangle.”
Set bailout to 8.
Edit user code:

```
public static Complex f9(Complex z) {

double x = z.getRe();
double y = z.getIm();
double x1 = 0;
double y1 = 0;

x1 = 2*x;
y1 = 2*y;
if (y > 0.5) {
        x1 = 2*x;
        y1 = 2*y-1;
}
if (x > 0.5)
{
        x1 = 2*x-1;
        y1 = 2*y;
}

return new Complex(x1, y1);

}
```

Run with “f9(z)”.

sqr(1/fn).

Set bailout to 8. "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter next to "z = ": (1/sin(z))^2 and press "ok".

sqr(fn).

Set bailout to 8. "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter next to "z = ": (sin(z))^2

tetrate.

Set bailout to 2. "fractal", "fractal options", "fractal functions", "user formulas", "user formula". Enter next to "z = ": c^z

```
public static Complex g10(Complex z, Complex c) {

 return c.pow(z);
 }
```

Call it with g10(z,c)

Unity

Set bailout to 1.5.
Via “edit user code”

```
public static Complex f7(Complex z) {

double x = z.getRe();
double y = z.getIm();
double one = (x*x)+(y*y);
double y1  = (2-one)*x;
double x1 = (2-one)*y1;

return new Complex(x1, y1);

}
```

Call with “f7(z)” in user formula.

Formula

flip3_man_m

z = 0:

Original forumula z = 1/flip(sqr(z) + pixel),
  |z| <= 4

Enter user formula next to z: “1/flip(z^2+c)”
Set bailout to 2.
“Options”, “Fractal Options”, “Initial value”. Enable “initial value”, next to real: “0”.

RCL_Pick1

Ron Lewen, 76376,2567

Try corners=2.008874/-3.811126/-3.980167/3.779833/-3.811126/3.779833 to see Figure 9.7 (P. 123) in Pickover's Computers, Pattern, Chaos and Beauty.

Original formula: z=cosh(z) + pixel

Enter user formula next to z: (cosh(z)-1+c)
Set bailout to 8.
“Options”,“bailout condition”, “User bailout condition”. Left operand: “abs(z)”. Right operand: “Bail”.

RCL_Pick10

Enter user formula next to z: (z/c-c*z^2)
Set bailout to 8.
“Options”,“bailout condition”, “User bailout condition”. Left operand: “abs(z)”. Right operand: “Bail”.

RCL_Pick3

Enter user formula next to z: (z/c-c*z^2)

Set bailout to 8.
“Options”,“bailout condition”, “User bailout condition”. Left operand: “abs(z)”. Right operand: “Bail”.
“Options”, “Fractal Options”, “Initial value”. Enable “initial value”, next to real: “0.5”

RCL_Pick4

Set bailout to 8.
Enter user formula next to z: (z*c-c*z^2)

Somethingelse

Original formula: z = 1:
  z = pixel * (z*z + 1/z/z),
  |z| <= 1000000

In user formula, next to z: c*(z*z+1/z/z).
“Options”, “Fractal Options”, “Initial value”. Enable “initial value”, next to real: “1”

TobeyRichard3 (XAXIS) {; Jm Richard-Collard

  ; Generalized by Tobey J. E. Reed
Original formula
z = pixel:
  sh=fn1(z), z=(1/(sh*sh))+pixel,
   |z|<=50

Set bailout to 7.
In user formula, next to z: “1/(sin(z)^2)+c”.

Zppchco8

Lee Skinner

Original formula:
z = pixel, f = cosxx (pixel):
z = cosh (z) + f,
|z|<=8192

Set bailout to 64.
In user formula, next to z: cosh(z)+conj(cos(c))

Fzpcocoh

Lee Skinner
Original formula:
z = pixel, f = 1. / cosh(pixel):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 16.
Enter the following user formula: next to “z =” “conj(cos(z))+1/cosh(c)”

Fzpcopch

Lee Skinner [75450,3631]
Original formula: z = pixel, f = pixel ^ (cosh(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 16.
Enter the following user formula: next to “z =” “conj(cos(z))+c^cosh(c)”

Fzpcopcs

Lee Skinner
Original formula:
z = pixel, f = pixel ^ (1. / cosxx(pixel))
z = cosxx (z) + f
|z| <= 50

Set bailout to 16.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(1/conj(cos(c)))”

Fzpcopct

Lee Skinner
Original formula: z = pixel, f = pixel ^ (cosxx(pixel) / sin(pixel)):
z = cosxx (z) + f,
|z|<= 50

Set bailout to 16.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(conj(cos(c))/sin(c))”

Fzpcophc

Lee Skinner
z = pixel, f = pixel ^ (1. / cosh(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(1/cosh(c))”

Fzpcophs

Lee Skinner

Original formula: z = pixel, f = pixel ^ (1. / sinh(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(1/sinh(c))”

Fzpcopht

Lee Skinner
Original formula: z = pixel, f = pixel ^ cotanh(pixel) :
z = cosxx (z) + f,
|z|<= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(coth(c))”

Fzpcopse

Lee Skinner
Original formula: z = pixel, f = pixel ^ (1. / sin(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(1/sin(c))”

Fzpcopsh

Lee Skinner
Original formula z = pixel, f = pixel ^ (sinh(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(sinh(c))”

Fzpcopsq

Lee Skinner
Original formula z = pixel, f = pixel ^ (sqr(pixel)):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(c^2)”

Fzpcopta

Lee Skinner
Original formula z = pixel, f = pixel ^ (sin(pixel) / cosxx(pixel)):
z = cosxx (z) + f,
|z|<= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(sin(c)/conj(cos(c)))”

Fzpcopth

Lee Skinner
z = pixel, f = pixel ^ tanh(pixel):
z = cosxx (z) + f,
|z|<= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “conj(cos(z))+c^(tanh(c))”

Fzpcoseh

Lee Skinner
z = pixel, f = 1. / sinh(pixel):
z = cosxx (z) + f,
|z| <= 50

Set bailout to 8.

Enter the following user formula: next to “z =” “conj(cos(z))+(1/sinh(c))”

Fzppchco

Lee Skinner
  z = pixel, f = cosxx (pixel):
  z = cosh (z) + f,
       |z| <= 50

Set bailout to 8.
Enter the following user formula: next to “z =” “cosh(z)+conj(cos(c))”

Wineglass

By Pieter Branderhorst
Enter the following user formula: next to “z =” “z^2+c” Next to “c = “ “(1+flip(im(c)))*re(c)/2+z”. Rotate 90°. Options → rotation → set rotation. Enter 90 above. Check next to “Current center”.

Mothra

By Ron Lewen
Remember Mothra, the giant Japanese-eating moth?
Well… here he (she?) is as a fractal!

“Options”, “Bailout condition”, “Real Strip”. Set bailout to 9 or 10. Set iterations to 500.

Enter as user formula, next to “z =” “(z^4+z^2+c)^2/(z^5+z^3+z+c)”.

Saved command set

These are variants of regular fractals (see chapter FractInt fractals) with certain parameters which were saved.

Filament

Extra information: A mandelbrot.
Author: Ian Adam
Fractal: mandel.

Start the regular "Mandelbrot z =z ^ 2 + C". “File” “Go to”, or simply press “F3”. Click on “set corners”.

For Corner 1 (top left):
Real: -1.860573964917644373961991277610650286078453063964843 75
Imaginary: -0.000000929480032675784847813406537586633504588462528772652149 2.

Corner 2 (Bottom right):
Real: -1.860573955176554439283620467904256656765937805175781 25
Imaginary: -0.000000939221122610463218623112931215946019847251591272652149 2

Change pallette (out) to “Legacy FractInt maps” and finally “Blue”.

Hypnoeyes2

Extra information: with decomp, try fast (color) cycling.
Author: Pieter Branderhorst
Fractal: Julia

Use the builtin "Mandelbrot z =z ^ 2 + C".

Use the following julia seeds:
Real: 0.25891900000000001025313167701824568212032318115234375
Imaginary:0.000000176951000000000007367339919552018390191960861557163298130035400 39

For Corner 1 (top left):
Real: -0.16245799999999999130295691429637372493743896484375
Imaginary: 1.02834249999999999269917339006497059017419815063476562 5

Corner 2 (Bottom right):
Real: 0.17486999999999999699085151405597571283578872680664062 5
Imaginary: 0.69101450000000003148770133520883973687887191777246093 75

Change pallette (out) to “Legacy FractInt maps” and finally “Blue”.

“Options”, “colors”, “out coloring mode”, “Color decomposition”.

Hypnoteyes

Same as Hypnoeyes2 but with “Chroma” color pallette.

Lambdafn

Author: Michael Coddington
Fractal: Lambdafn

Set interations really high. (e.g. 5000). Set bailout to 16. Enter the following user formula:
(4.72675+0.00145555i)*cos(z)

“Go to”

Real:
0.000163735795919703965680977175230048793267184071781772931198218554540393620853390968333497

Imaginary:
0.001840390346137472574254183449585748436323148966827127746667976553034024298392114484068507

Size: 5.267489711934156378600823045267489711934156378600823045267489711934156378600823045267489e-1

Click on “palette (out)”, “Legacy/FractInt Maps”, “Lambdafn”

Minimandelbrot

Author: Pieter Branderhorst
Fractal: Mandel

Start the original the original Mandelbrot under "Mandelbrot type" and "Mandelbrot z =z ^ 2 + c".

Click on “File”, “Go to”. Click on corners en enter following values:

Real: -0.74542849510500005
Imaginary: -0.11300864443
Size: 2.08504499999e-5

Click on “palette (out)”, “Legacy/FractInt Maps”, “Minimandelbrot”

Rich8z3

Author: Lee Skinner
Fractal: Fn(z)+Fn(pix)
Fractal Creations cover

Set iterations to 1000,
Enter the following user formula: sin(z)+sin(c)
“Go to”: Real: 0.58610439 Imaginary: 1.17509517 Size: 9.1504e-3

Click on “palette out”, “Legacy/FractInt maps”, “Rich8z3”.

Spiderplant

Author: Richard Hughes
Fractal: manowarj

In fractal zoomer this is called "Manowar".

Click "Tools", "julia" of simply press "J". Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0" Imaginary "0".

“Go to”
Real: 0.14510172684119687636083995130320545285940170288085937 5
Imaginary: 0.05800615729822721639008520355673681478947401046752929687 5
Size: 7.6187699999999955458029887722659623250365257263183593 75e-2

“Palette (Out)”, “Legacy/FractInt maps”, “blues”.

Options → rotation → set rotation. Set rotation to 30°. Check next to “Current center”.

YinFinite

Like Yin-Yang symbol, infinity sign

Author: Ethan Nagel
Fractal: Julia

Set iterations high, e.g. 5000.

Use the builtin Mandelbrot z^2 +c. "Tools", "Julia" (or simply "J" on keyboard). Hit "File", "Go to" or simply "Ctrl F3" on keyboard. Enter the following (julia seed) values: Real "0.252235" Imaginary "0.000169836".

Again “Go to”
Real of new center: -0.0033692
Imag of new center: -0.00053805
Size: 2.589626e-1