A Mandelbrot Set Explorer

GONZALO GARCÍA ALARCÓN ESTRADA, 1004229676
Jan / 07 / 2018
University of Toronto, Shameless Mathematica

USING THE ESCAPE TIME ALGORITHM

```
(* for saving data where the Notebook is *)
NotebookDirectory[] // SetDirectory

(* Export["Mandelbrot_counter",{1},"List"] *) (* reset counter!!!!! *)
```

```
C:\drorbn\AcademicPensieve\Classes\17-1750-ShamelessMathematica\StudentProjects\
  garcia_gonzalo_180107_mandelbrot
```

```
f_c_[z_] := z^2 + c
```

```
(* Escape Time Algorithm *)

EscTime[c_] := Module[{i = 1, z = 0},
  While[(Abs[z] < 4) ∧ (i < ilim),
   z = f_c[z];
   i++];
  Return[i]
 ]
```

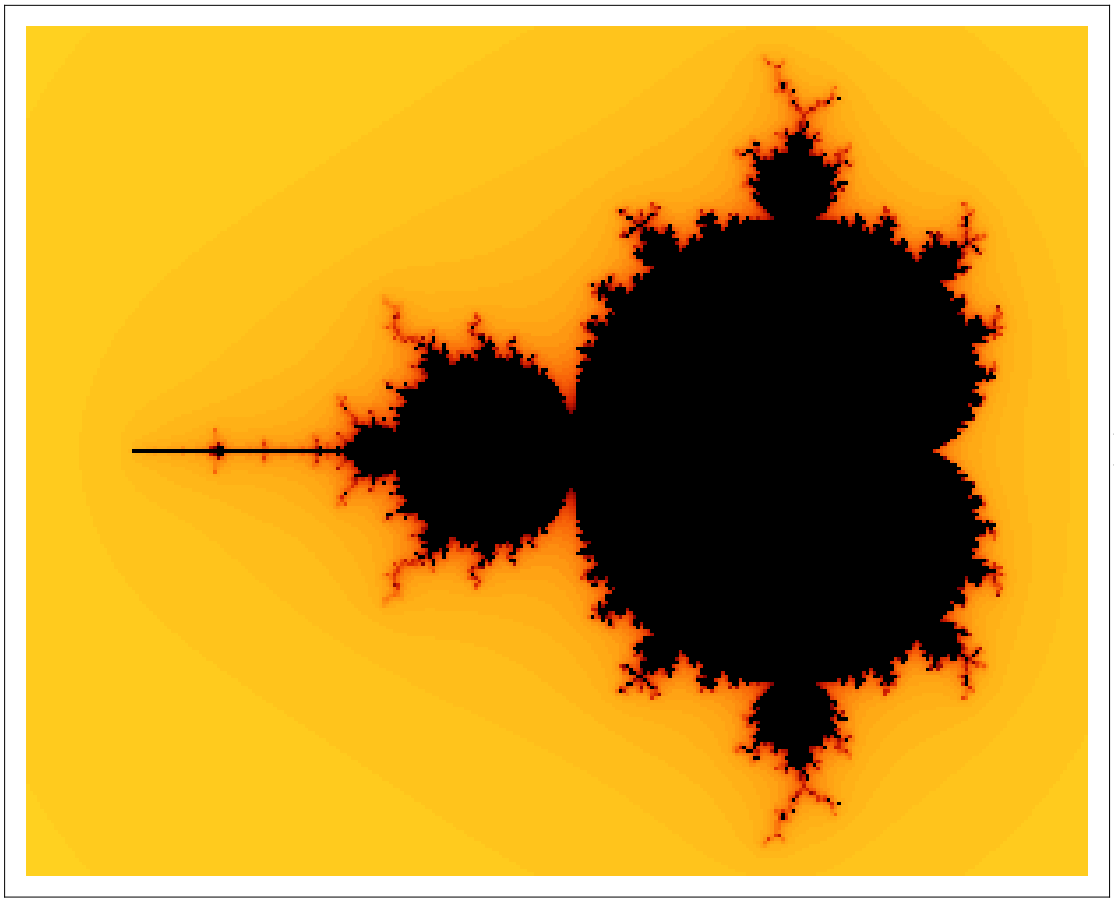
```
(* PLAY WITH THIS ONE *)

xmin = -2.3;
xmax = 0.7;
ymin = -1.2;
ymax = 1.2;
Δ = 0.01; (* pixel width *)
ilim = 30; (* refines border detection *)
```

```
{t, M} = Timing[
   Table[{x, y}, {y, ymin, ymax, Δ}, {x, xmin, xmax, Δ}] /. {x_, y_} :> EscTime[x + i y]];
t
```

```
2.60938
```

```
plot = ArrayPlot[M, ColorFunction → ColorData[{"SolarColors", "Reverse"}],
    ColorRules → {ilim → Black}, ImageSize → Large] // Timing
```

```
{0.171875,
```

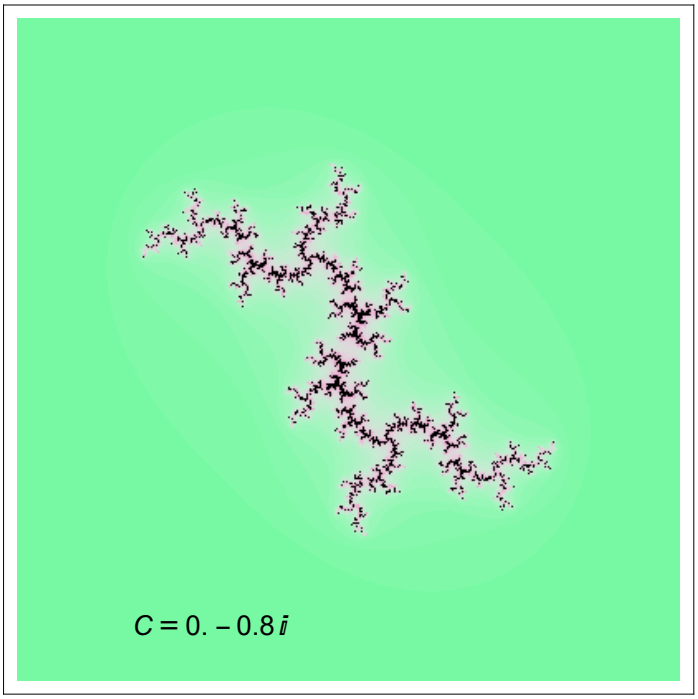


```
}
```

ESCAPE TIME ALGORITHM - for JULIA SET

```
(* Escape Time Algorithm *)

EscTimeJulia[z_, c_] := Module[{i = 1, F = fc[z]},
  While[(Abs[F] < 2) ∧ (i < ilim),
   F = fc[F];
   i++];
  Return[i]
 ]
```

```
(* PLAY WITH THIS ONE *)

xminJ = -2;
xmaxJ = 2;
yminJ = -2;
ymaxJ = 2;
ΔJ = 0.01; (* pixel width RESOLUTION *)
ilimJ = 100; (* refines border *)

(* some examples to play with *)
c = 0;
c = -2;
c = -0.70176 - 0.3842 i;
c = 0.285;
c = -0.4 + 0.6 i;
c = -0.8 i;

{t, MJ} = Timing[Table[{x, y}, {y, yminJ, ymaxJ, ΔJ}, {x, xminJ, xmaxJ, ΔJ}] /.
    {x_, y_} :> EscTimeJulia[x + i y, c]];

plot = ArrayPlot[MJ, ColorFunction → ColorData["MintColors"],
  ColorRules → {Max[MJ] → Black}, ImageSize -> Medium,
  Epilog → Text[Style[C == c, FontSize → Scaled[0.04]], Scaled[{.3, .1}]]]
t
```

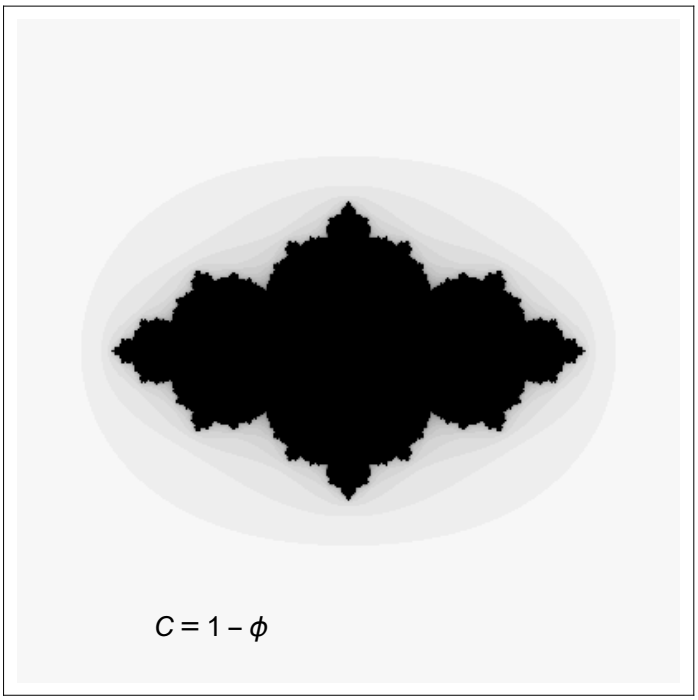



```
2.39063
```

MANDELBROT EXPLORER:

```
JuliaPlot[c_] := Module[{xmin = -2, xmax = 2, ymin = -2, ymax = 2, Δ = 0.01, ilim = 20},

  {t, M} = Timing[Table[{x, y}, {y, ymin, ymax, Δ}, {x, xmin, xmax, Δ}] /.
     {x_, y_} :> EscTimeJulia[x + i y, c]];

  plot = ArrayPlot[M, ImageSize -> Medium,
    Epilog → Text[Style[C == c, FontSize → Scaled[0.04]], Scaled[{.3, .1}]]]

 ]
```

```
JuliaPlot[1 - GoldenRatio]
```




```
color = ColorData[{"SolarColors", "Reverse"}];
colorset[p_] := If[p == 1, Black, color[p]]
```

```
(* repeated the Mandelbrot set.... in case parameters were changed *)
xmin = -2.3; xmax = 0.7; ymin = -1.2; ymax = 1.2; Δ = 0.006;   ilim = 30;
```

```
Mexpl = Table[{x, y}, {y, ymin, ymax, Δ}, {x, xmin, xmax, Δ}] /. {x_, y_} :> EscTime[x + i y];
i1 =
  Image[Map[colorset, (Mexpl - Min[Mexpl]) / (ilim - Min[Mexpl]), {2}], ImageSize -> Large];
```

```
(* from pixel to complex plane *)
PxC[x_] := Δ x + xmin - Δ / 2;
PyC[y_] := Δ y + ymin - Δ / 2;

(* from complex plane to pixel *)
CxP[x_] := (x - xmin) / Δ + 1 / 2;
CyP[y_] := (y - ymin) / Δ + 1 / 2;

l = Graphics[{Cyan, Circle[{0, 0}, 2], Point[{0, 0}]}, ImageSize → 20]
```

```
(* MANDELBROT EXPLORER..... be patient....it'
 s nice. Had to sacrifice resolution for speed *)
(* delete the cell after using to avoid your computer becoming extra slow *)

Manipulate[
 {Show[{
    i1,
    Graphics[{
      Text[
        Style[Re[C] == PxC[p[[1]]], FontSize → Scaled[0.03]], Scaled[{.2, .9}]] // Dynamic,
      Text[Style[Im[C] == PyC[p[[2]]], FontSize → Scaled[0.03]], Scaled[{.2, .85}]] //
       Dynamic, {White, Line[{{{xmin // CxP, 0 // CyP}, {xmax // CxP, 0 // CyP}},
          {{0 // CxP, ymin // CyP}, {0 // CxP, ymax // CyP}}}]}
     }]
   }], Style[Dynamic[JuliaPlot[c]], DynamicEvaluationTimeout → 60]},
 {{p, {CxP[0], CyP[0]}}, Locator, Appearance → l}, {{c, 0}, None},
 Button["Plot Filled-in Julia set", c = PxC[p[[1]]] + i PyC[p[[2]]] ]
]
```

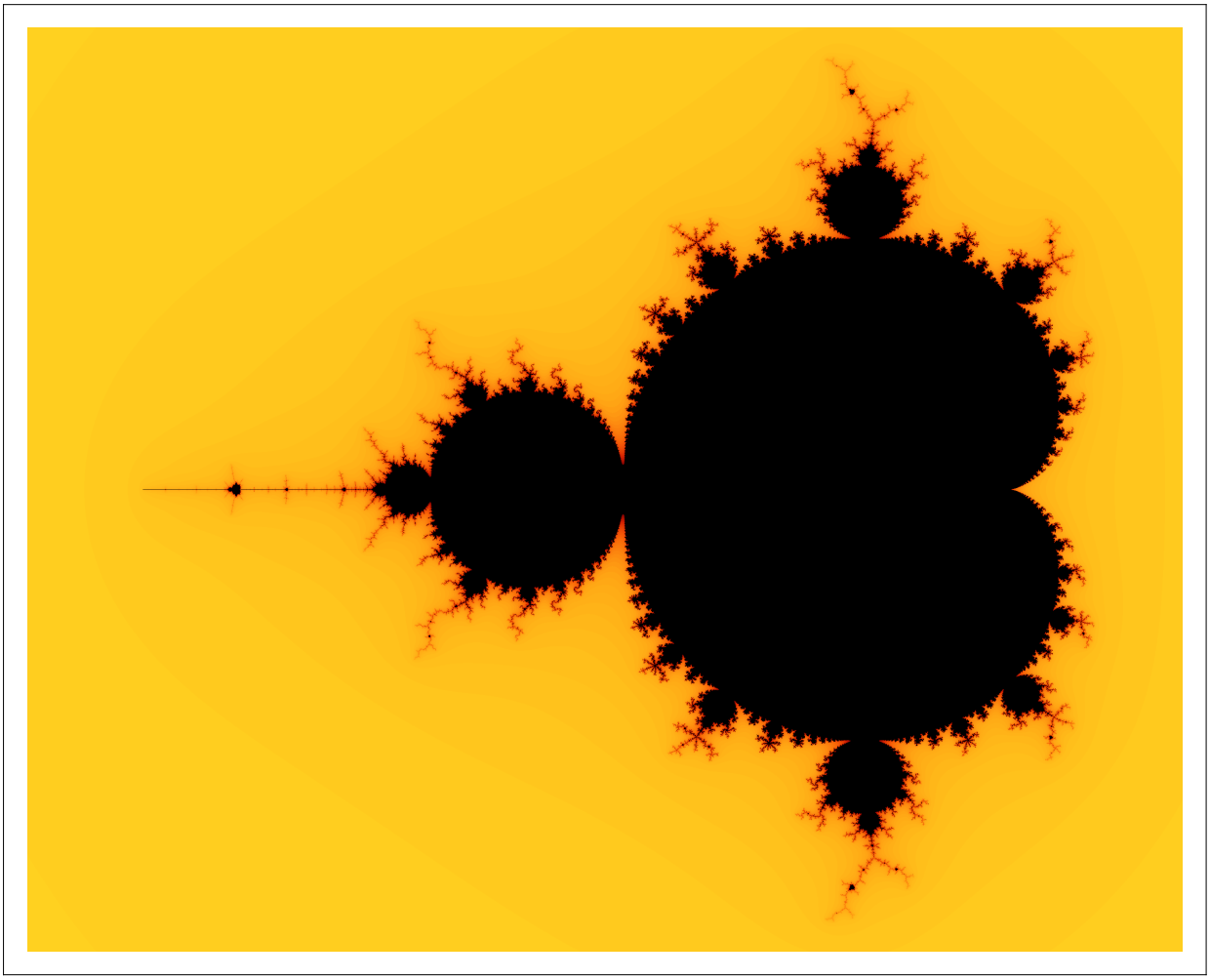
HD IMAGES ... do not run! takes a long long time!

```
xmin = -2.3;
xmax = 0.7;
ymin = -1.2;
ymax = 1.2;
Δ = 0.001; (* pixel width *)
ilim = 50; (* refines border *)
```

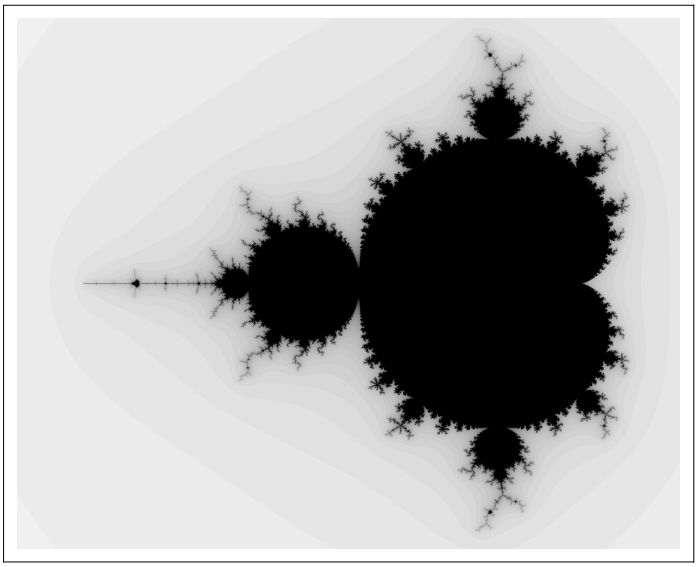
```
(* !!!!!!!!!!!!!!!!!  DO NOT RUN!!!! TAKES A LONG TIME  !!!!!!!!!!!!!!!!!  *)

{t, M} = Timing[
   Table[{x, y}, {y, ymin, ymax, Δ}, {x, xmin, xmax, Δ}] /. {x_, y_} :→ EscTime[x + i y]];
t
```

```
1681.29
```

```
plot = ArrayPlot[M, ColorFunction → ColorData[{"SolarColors", "Reverse"}],
  ColorRules → {ilim → Black}, ImageSize → Full]
```



```
(* here is the HD array stored *)
Mhd = Import["MandelbrotHD_Data.dat", "Table"];
Mhd // ArrayPlot
```



Saving procedures:

```
filename = "MandelbrotEscapeTime";
fileindex = Import["Mandelbrot_counter", "List"][[1]];
filename = filename <> "_" <> ToString[fileindex]
```

```
MandelbrotEscapeTime_4
```

```
Export[filename <> "_Data.dat", M, "Table"]
Export[filename <> "_Plot.png", plot]
Export["Mandelbrot_counter", {fileindex + 1}, "List"]
```

```
MandelbrotEscapeTime_4_Data.dat
```

```
MandelbrotEscapeTime_4_Plot.png
```

```
Mandelbrot_counter
```