

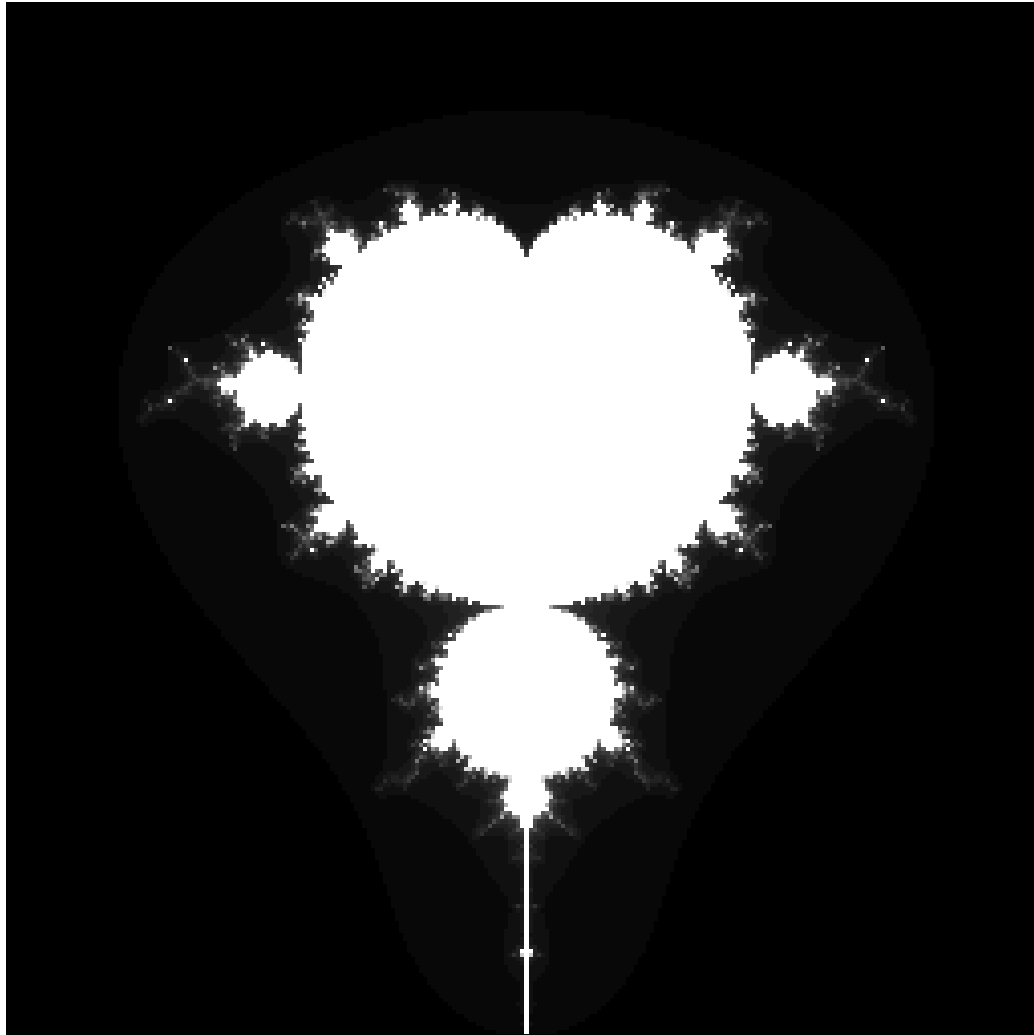
```
(* Using a 1D Texture Gauge in a MandelBrot Set for Self-Coloring Fractals
Copyright Doug Youvan on April 15,
2006 www.youvan.com and www.pseudocolor.com *)
```

```
xclow = -2.0;
xchi = 1.0;
yclow = -1.5;
ychi = 1.5;
numxpoints = 256;
numypoints = 256;
numits = 100;
xcrange = xchi - xclow;
ycrange = ychi - yclow;
incxc = xcrange / numxpoints;
incyc = ycrange / numypoints;

(* Draw Mandelbrot *)
fracmat = Table[{x, y}, {x, 1, numxpoints}, {y, 1, numypoints}];
fracmat[[All, All]] = {0, 0, 0};
For[x = 1, x ≤ numxpoints, x++,
  xcvar = xclow + ((x - 1) * incxc);

  For[y = 1, y ≤ numypoints, y++,
    ycvar = yclow + ((y - 1) * incyc);
    z = 0;
    c = xcvar + (ycvar * I);
    For[i = 1, ((i ≤ numits) && (Abs[z] < 2)), i++, z = z^2 + c;
      fracmat[[x, y]] = {i / numits, i / numits, i / numits}]]];
```

```
In[84]:=  
fracgraph1 = RasterArray[Apply[RGBColor, fracmat, {2}]];  
Show[Graphics[fracgraph1],  
ImageSize -> {2 * numxpoints, 2 * numypoints}, AspectRatio -> Automatic];
```

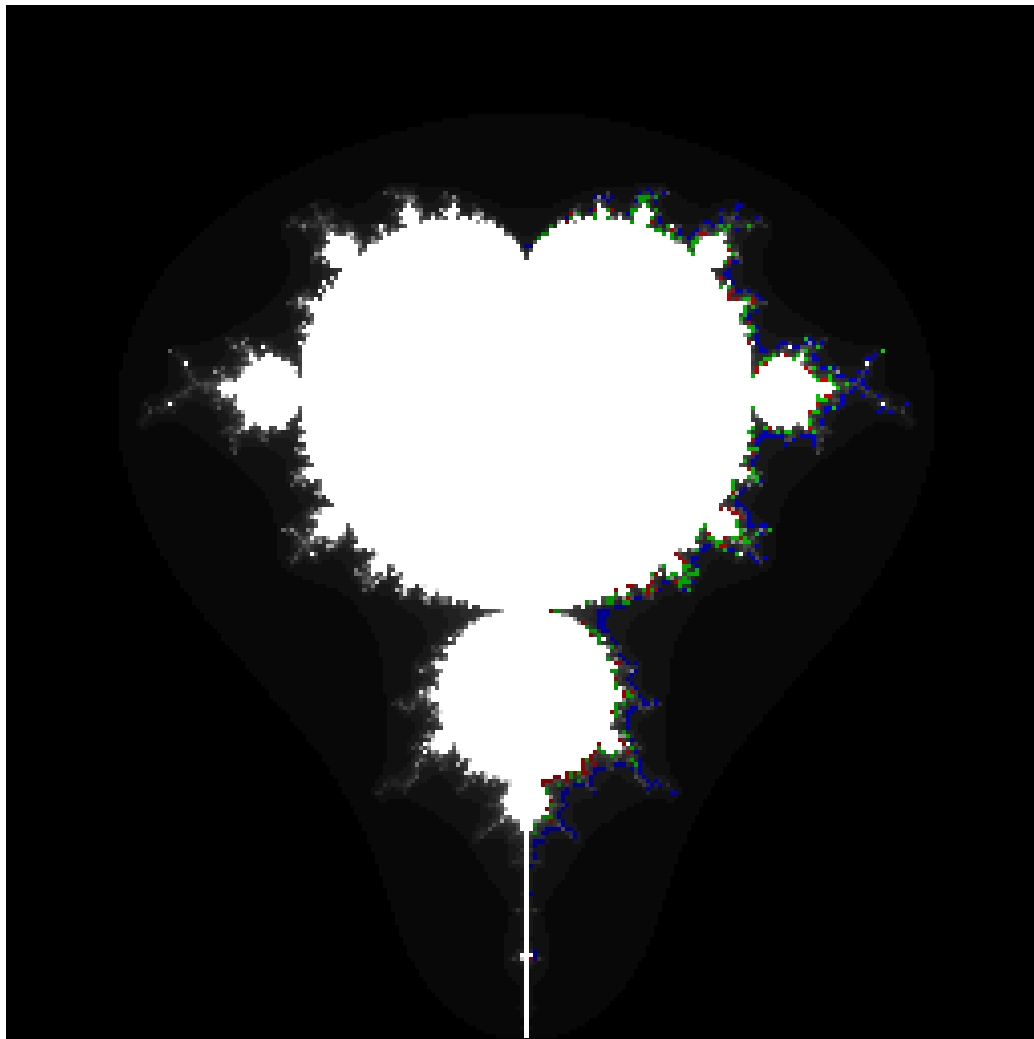


```

In[86]:= (* Depth Gauge and Mandelbrot rendering of depth *)
(* Show Graph has x,y = 0,0 at lower left,
   however in MatrixForm x,y=0,0 is upper,left ???*)
dep = Table[{x}, {x, 1, numxpoints}];
dep[[All]] = {0, 0, 0};
centery = Floor[numypoints / 2] + 1;
For[x = 1, x ≤ numxpoints, x++, r = 1; g = 1; b = 1;
  For[y = centery, y ≤ 2 * (centery - 2), y++,
    If[(0.6 < fracmat[[x, y, 1]]) && (fracmat[[x, y, 1]] < 1.0),
      r = y/centery / 2; dep[[x]] = {r, 0, 0}; fracmat[[x, y]] = {r, 0, 0},
    If[(0.3 < fracmat[[x, y, 1]]) && (fracmat[[x, y, 1]] < 0.5),
      g = y/centery / 2; dep[[x]] = {r, g, 0}; fracmat[[x, y]] = {0, g, 0},
    If[(0.12 < fracmat[[x, y, 1]]) && (fracmat[[x, y, 1]] < 0.15),
      b = y/centery / 2; dep[[x]] = {r, g, b}; fracmat[[x, y]] = {0, 0, b} ]]]];

Show[Graphics[RasterArray[Apply[RGBColor, fracmat, {2}]],
  AspectRatio → error, ImageSize → {2 * numxpoints, 2 * numypoints}]];

```

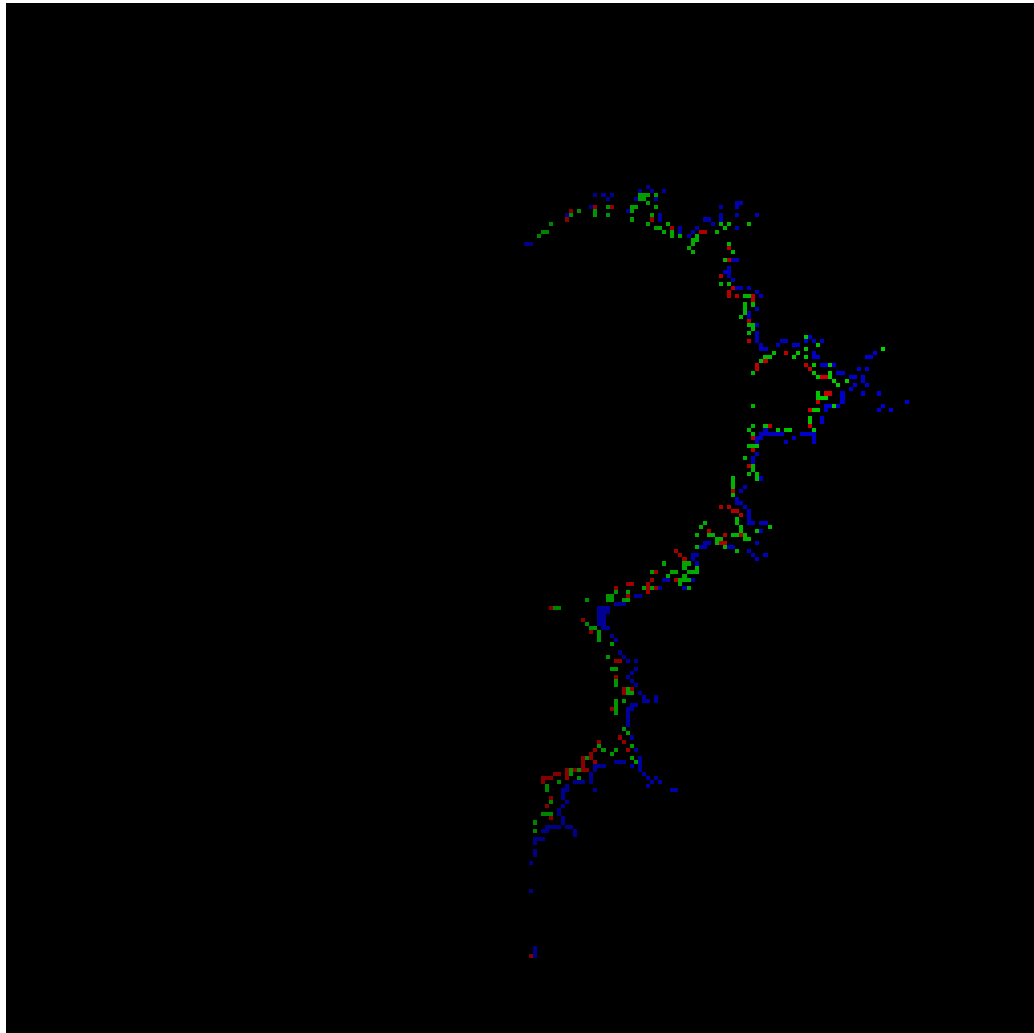


```
In[91]:=
(* duplicate colorized fracmat into frac *)

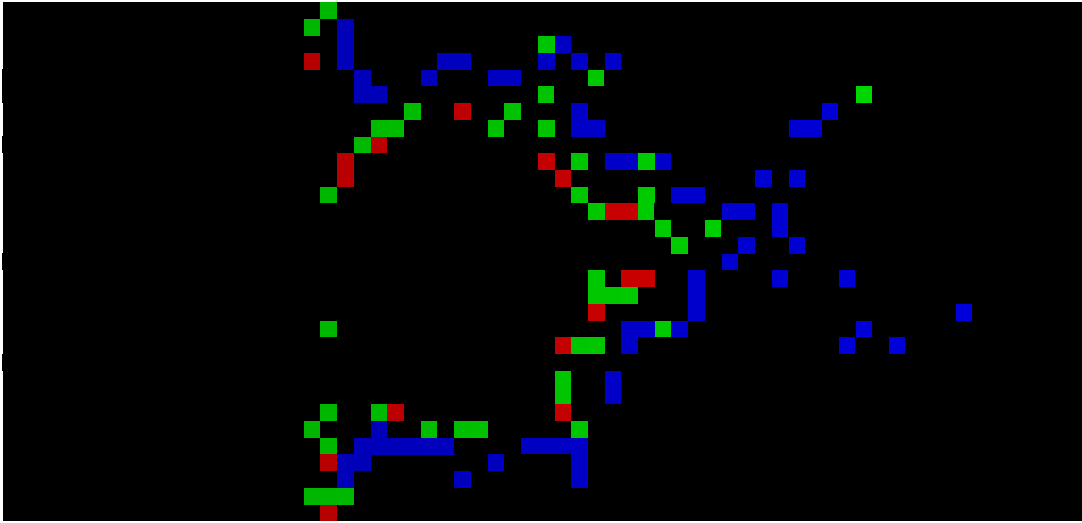
frac = Table[{x, y}, {x, 1, numxpoints}, {y, 1, numypoints}];
frac[[All, All]] = {1, 1, 1};
For[x = 1, x ≤ numxpoints, x++,
  For[y = 1, y ≤ numypoints, y++,
    frac[[x, y]] = fracmat[[x, y]] ]];

(* Eliminate gray pixels in depth rendering *)

For[x = 1, x ≤ numxpoints, x++,
  For[y = 1, y ≤ numypoints, y++,
    If[frac[[x, y, 1]] == frac[[x, y, 2]] == frac[[x, y, 3]], frac[[x, y]] = {0, 0, 0} ]];
Show[Graphics[RasterArray[Apply[RGBColor, frac, {2}]],
  AspectRatio → error, ImageSize → {2 * numxpoints, 2 * numypoints}]];
```



```
In[96]:= (* zoom *)  
Show[Graphics[RasterArray[Apply[RGBColor, frac, {2}]], AspectRatio -> Automatic,  
PlotRange -> {{165, 230}, {145, 176}}, ImageSize -> {512, 245}]];
```



```
In[97]:= (* generate color bar *)
pcb = Table[{x, #}, {x, 1, numxpoints}, {y, 1, 10}];
For[x = 1, x ≤ numxpoints, x++,
  pcb[[x, All]] = dep[[x]] ];

Show[Graphics[RasterArray[Apply[RGBColor, pcb, {2}]],
  ImageSize → {30, 2 * (numxpoints + 10)}, AspectRatio → Automatic];
```



```
In[102]:=
```

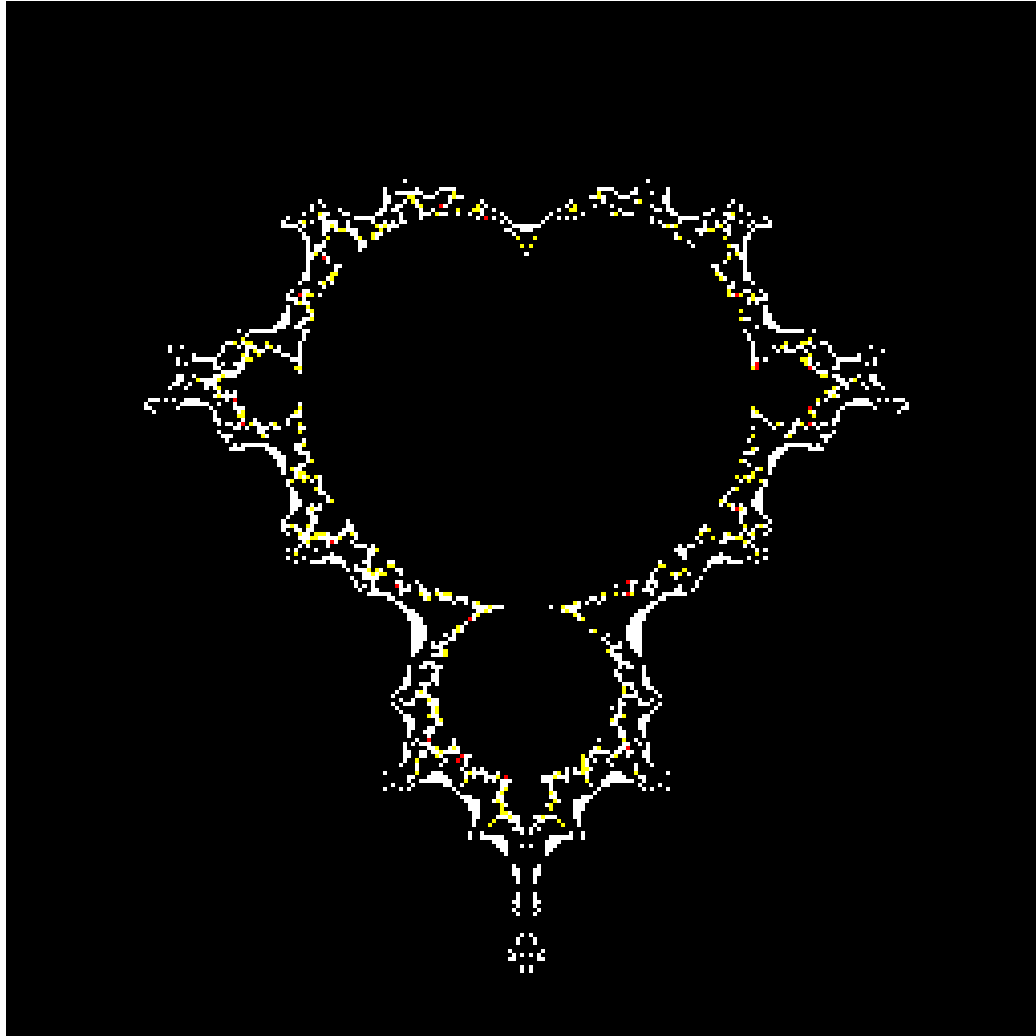
```
(* rescale pcb from 0-1 to 1-256 for indexing *)
pcbindex = Table[{x}, {x, 1, numxpoints}];
pcbindex[[All]] = {0, 0, 0};

For[x = 1, x ≤ numxpoints, x++,
  pcbindex[[x]] = Floor[255 * pcb[[x, 5]]]; (*was 256*)
  (*index fractal grayscale into fractal pseudocolor scheme from above *)
  colorfrac = Table[{x}, {x, 1, numxpoints}, {y, 1, numypoints}];
  colorfrac[[All, All]] = {0.5, 0.5, 0.5};

  (* change grayscale of fractal from 0-1 to 1-255 for indexing *)
  For[x = 1, x ≤ numxpoints, x++,
    For[y = 1, y ≤ numypoints, y++,
      colorfrac[[x, y, 1]] = Floor[255 * (fracmat[[x, y, 1]])]; (*255 or 256! *)
      If[colorfrac[[x, y, 1]] ≤ 1, colorfrac[[x, y, 1]] = 1];
    If[colorfrac[[x, y, 1]] > 255, colorfrac[[x, y, 1]] = 255];
    (* only for r channel which is now gray *)

  (* Display Self-Colored Fractal *)
  (* There is a minor bug in the indexing that results in
  some full RGB values leaking through; these values are suppressed
  by Show[Graph]. If found, the bug fix will be updated *)

  For[x = 1, x ≤ numxpoints, x++,
    For[y = 1, y ≤ numypoints, y++,
      colorfrac[[x, y]] = pcbindex[[colorfrac[[x, y, 1]]]];
    Show[Graphics[RasterArray[Apply[RGBColor, colorfrac, {2}]],
      AspectRatio → Automatic, ImageSize → {2 * numxpoints, 2 * numypoints}]]];
```



```

In[110]:=
(* repeat the above (verbatim, except for the <<<<<<<<< line for zoom *)

colorfrac = Table[{x}, {x, 1, numxpoints}, {y, 1, numypoints}];
colorfrac[[All, All]] = {0.5, 0.5, 0.5};

For[x = 1, x ≤ numxpoints, x++,
  For[y = 1, y ≤ numypoints, y++,
    colorfrac[[x, y, 1]] = Floor[255 * (fracmat[[x, y, 1]])]; (*255 or 256! *)
    If[colorfrac[[x, y, 1]] ≤ 1, colorfrac[[x, y, 1]] = 1];
  If[colorfrac[[x, y, 1]] > 255, colorfrac[[x, y, 1]] = 255];

For[x = 1, x ≤ numxpoints, x++,
  For[y = 1, y ≤ numypoints, y++,
    colorfrac[[x, y]] = pcbindex[[colorfrac[[x, y, 1]]]];
  Show[
  Graphics[RasterArray[Apply[RGBColor, colorfrac, {2}]], AspectRatio → Automatic,
  PlotRange → {{165, 230}, {145, 176}}, ImageSize → {512, 245}]];
(*<<<<<<<<< Zoom area of interest*)

```

