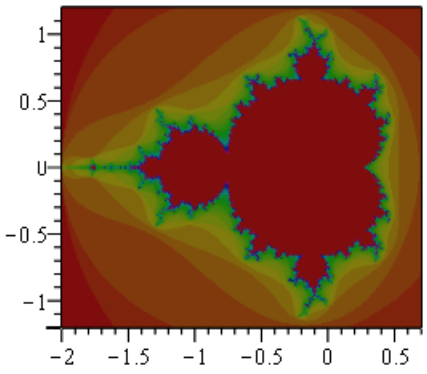
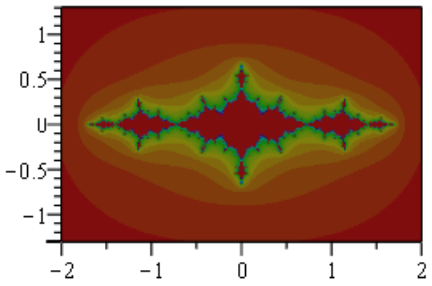
```
restart : with(plots) :

 mandelbrot := proc(x, y)
   local c, z, m;
   c := evalf(x + y*I);
   z := c;
   for m from 0 to 30 while abs(z) < 2 do
     z := z^2 + c
od;
   m
 end:

plot3d(0, -2 .. 0.7, -1.2 .. 1.2, orientation = [-90, 0], grid = [250, 250],
   style = patchnogrid, scaling = constrained, color = mandelbrot);
```




```
 julia := proc(c, x, y)
   local z, m;
   z := evalf(x + y*I);
   for m from 0 to 30 while abs(z) < 3 do
     z := z^2 + c
```

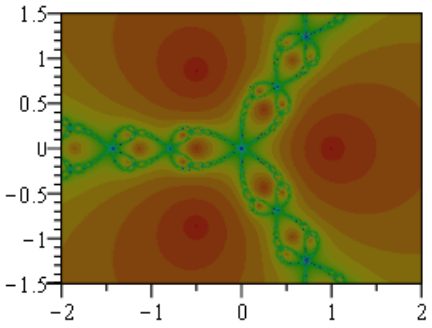
```
  od;
   m
end:
J := proc(d)
   global phonyvar;
   phonyvar := d;
   (x, y) → julia(phonyvar, x, y)
end:
plot3d(0, -2 .. 2, -1.3 ..1.3, style=patchnogrid,
   orientation=[-90, 0], grid=[250, 250],
   scaling=constrained, color=J(-1.25));
```




```
solve(z^3-1=0, z);
newton := proc(x, y)
   local z, m;
   z := evalf(x+y*I);
  for m from 0 to 50 while abs(z^3-1) ≥ 0.001 do
     z := z - (z^3-1)/(3*z^2)
   od;
   m
end:
```

```
plot3d(0, -2 .. 2, -1.5 .. 1.5, orientation = [-90, 0],
  grid = [250, 250], style = patchnogrid, scaling = constrained,
  color = newton);
```

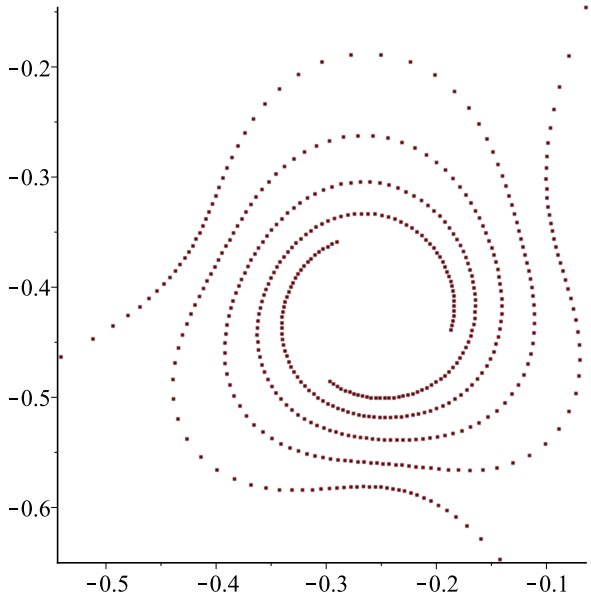
$$1, -\frac{1}{2} - \frac{1}{2} I\sqrt{3}, -\frac{1}{2} + \frac{1}{2} I\sqrt{3}$$

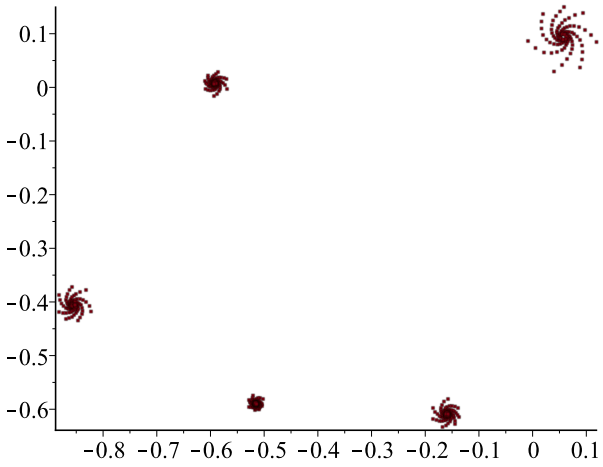


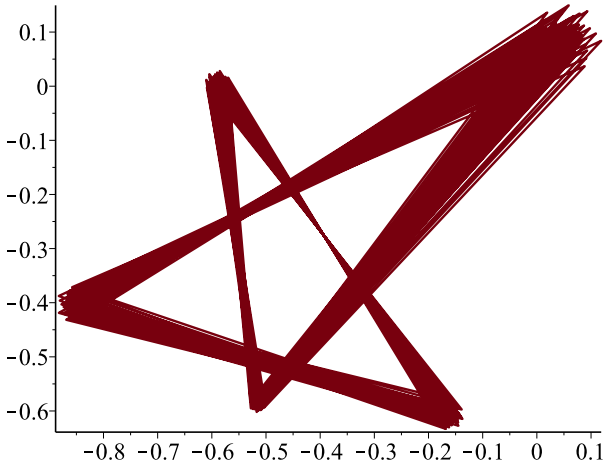

```
orbits := proc(x, y, iter)
 local c, z, pts;
 pts := NULL;
  c := evalf(x + y*I);
  z := c;
  to iter do
    if abs(z) < 1e10 then
      pts := pts, [Re(z), Im(z)]
   fi;
    z := z^2 + c
  od;
  [pts]
```

end:

```
plot(orbits(-0.14222874119878, -0.64732701703906, 500), style=point,
   symbol=POINT, axes=framed, scaling=constrained);
plot(orbits(-0.50977517291904, -0.60039090737700, 500), style=point,
   symbol=POINT, axes=framed, scaling=constrained);
plot(orbits(-0.50977517291904, -0.60039090737700, 300), style=line,
  axes=framed, scaling=constrained);
```




0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.8 −0.7 −0.6 −0.5 −0.4 −0.3 −0.2 −0.1 0 0.1


0.1
0
−0.1
−0.2
−0.3
−0.4
−0.5
−0.6
−0.8 −0.7 −0.6 −0.5 −0.4 −0.3 −0.2 −0.1 0 0.1