

```

include("functions.jl")

ITERATIONS = 100
DENSITY = 1000
x_min, x_max = -2.68, 1.32
y_min, y_max = -1.5, 1.5

x,y = meshgrid( LinRange(x_min, x_max, DENSITY),
                LinRange(y_min, y_max, DENSITY))

c = x + (y)im
z = copy(c)
fractal = fill(255,DENSITY,DENSITY)

for n in 1:ITERATIONS
    println("Iteration ", n)
    mask = [abs(α) ≤ 10 ? true : false for α in z]
    z[mask] .^= 2
    z[mask] .+= c[mask]
    mask1 = [abs(α) > 10 ? true : false for α in z]
    mask2 = [ β == 255 ? true : false for β in fractal]
    fractal[mask1 .&& mask2] .= Int(floor( (254 * (n-1) / ITERATIONS) ))
end

logfractal = [ log(y) for y in fractal ]

println("Saving...")
savetxt("fractal_jl.dat", logfractal)
savetxt("coord_jl.dat", [ x_min, x_max, y_min, y_max ] )

exit()
#
# functions.jl
#

# meshgrid function by Chris Rackauckas, 2016
# https://groups.google.com/g/julia-users/c/83Pfg9HGhGQ/m/9G\_0wi-GBQAJ
function meshgrid(a,b)
    x = a' .* ones(length(b))
    y = ones(length(a))' .* b
    return x,y
end

function savetxt(filename::String, array::AbstractArray)
    io = open(filename, "w")
    for row in eachrow(array)
        line = replace(string(row), '[' => "", ']' => "", ',' => "")
        println(io, line)
    end
    close(io)
end

```