**Problem 3, Part D (finding/graphing a 2-cycle)**

Initial data:

(used code from class, changing bounds and equations to fit the context of this problem)

# make a cobweb plot

import numpy as np

import matplotlib.pyplot as plt

# here is the function f=func we want to iterate. corresponds to x\_{n+1}=f(x\_n)

# r is a possible parameter

def func(x,r):

return (3\*x - (x\*\*3))

# return f^n(x)

def func\_n(x,r,n):

for i in range(0,n):

x = func(x,r)

return x

# here is "plotting graphical" or "cobweb" for an interated map

# connect up (x, f^1(x)), (f^1(x),f^1(x)), (f^1(x), f^2(x)), (f^2(x),f^2(x))

# ... (f^i(x), f^(i+1)(x)),(f^(i+1),f^(i+1)) to i=n

# initial x0, r is a parameter to pass to function

# connect up points n times, this is 2n pairs of points

def plot\_graphical(x0,r,n):

xv = np.linspace(0.0,1.0,2\*n) # create array for points xvalue

yv = np.linspace(0.0,1.0,2\*n) # create array for points yvalue

x =x0

for i in range(0,n): #iterate

xv[2\*i] = x # first point is (x,f(x))

x = func(x,r)

yv[2\*i] = x

xv[2\*i+1] = x #second point is (f(x),f(x))

yv[2\*i+1] = x

plt.plot(xv,yv,'b') # connect up all these points blue

plt.figure()

plt.xlabel('x')

plt.ylabel('f(x)')

fac=1.01

xmax = 4.00

xmin =-4.00

ymax = 4.00

ymin =-4.00

plt.axis([xmin\*fac,xmax\*fac,ymin\*fac,ymax\*fac])

xcon = np.arange(xmin, xmax, 0.01) # to plot function

plt.plot(xcon,xcon, 'g') #y=x plotted green

r=2 #set r

x0=2.0 #set starting point

ycon = func(xcon,r) # function computed

plt.plot(xcon,ycon, 'r') # function plotted red

plt.title("x0= {}, r= {}".format(x0, r))

plot\_graphical(x0,r,50) # cobweb plot