import networkx as nx

import matplotlib.pyplot as plt

import random as r

def disp(g,ne,ind):

if ne=='':

ne=[]

pos=nx.circular\_layout(g)

node=g.nodes()

old\_edge=list(set(g.edges())-set(ne))

nx.draw\_networkx\_nodes(g,pos,nodelist=node,node\_color="r")

nx.draw\_networkx\_edges(g,pos,edgelist=old\_edge,edge\_color="blue")

nx.draw\_networkx\_edges(g,pos,edgelist=ne,edge\_color="g",style="dashdot")

plt.savefig(str(ind))

plt.close()

def erdos(g,p):

con\_edge=0

for i in g.nodes():

for u in g.nodes():

if i!=u:

val=r.random()

if (val<=p):

flag=g.has\_edge(i,u)

if flag==True:

continue

else:

g.add\_edge(i,u)

con\_edge+=1

disp(g,[(i,u)],con\_edge)

print 'Connected edges are: ',con\_edge

def main():

'''n=input('Enter the value of nodes:')

p=input('Enter the value of probability:')'''

n=40

p=0.2

g=nx.Graph()

for i in range(n):

g.add\_node(i)

disp(g,'',-1)

erdos(g,p)

main()