Home work2: Random Network

author: Kunlun Zhu

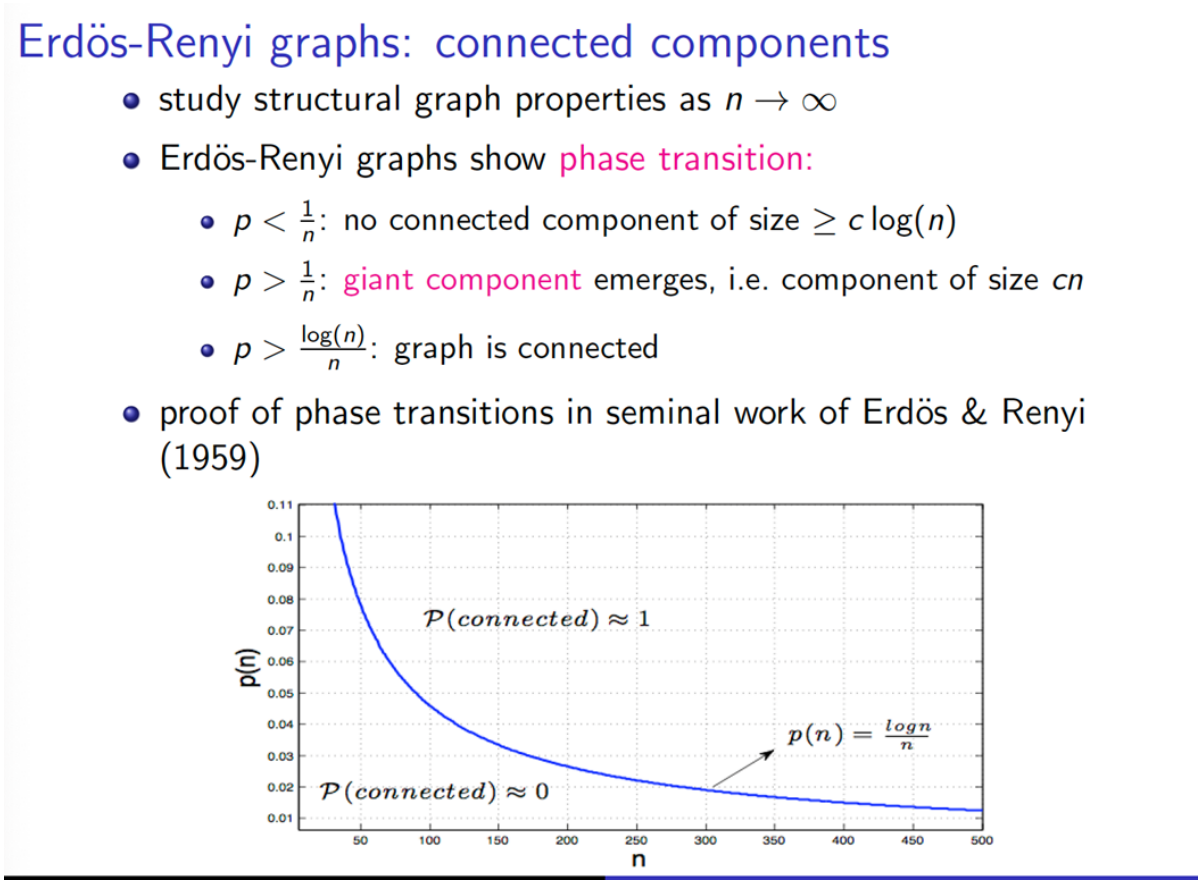
- Description: Generate a random ER network by varying p. Compute the size connected components. Plot that curve.

Step1: setup the environment

In [1]:
```
#!/usr/bin/python3.9
# -*- coding: utf-8 -*-
#author Kunlun Zhu 2022/7/01
#setup the python environment

import numpy as np
import networkx as nx
import matplotlib.pyplot as plt
```

Step2: Initialize the parameter With 100 node size and consider the following slide we intialize the p_list in this way
we have 1/n = 0.01, log(n) / n = 0.02



In [2]:
```
node_size = 100
p_list = [0, 0.001, 0.002, 0.003, 0.004, 0.005, 0.006, 0.007, 0.008, 0.009, 0.01,0.011
max_components_size = []
components_num = []
```

In [3]:
```
def gen_graph(G):#sub_graph generation for iteration
    for g in nx.connected_components(G):
```

```
        yield G.subgraph(g)

for p in p_list:
    ER_p = nx.random_graphs.erdos_renyi_graph(node_size, p)
    max_size = 0
    comp_num = 0
    for sub_g in gen_graph(ER_p):
        comp_num += 1
        if len(sub_g) > max_size:
            max_size = len(sub_g)
    max_components_size.append(max_size)
    components_num.append(comp_num)

print('max_components_size:', max_components_size)
print('components_num for each p:', components_num)
```

```
max_components_size: [1, 2, 3, 3, 5, 7, 6, 17, 19, 17, 13, 14, 42, 30, 48, 61, 69, 72, 74, 74, 86, 95, 97, 100, 100, 100]
components_num for each p: [100, 93, 89, 86, 79, 66, 77, 69, 53, 50, 47, 48, 40, 43, 28, 27, 24, 23, 25, 19, 13, 6, 4, 1, 1, 1]
```

In [4]:

```
print("Blue:Curve for max componets size with different p")
plt.plot(p_list, max_components_size)
print("Orange:Curve for components num for differnt p:")
plt.plot(p_list, components_num)
plt.xlabel('p')
plt.ylabel('num')
plt.title("Curve for ER graph with different p")
```

```
Blue:Curve for max componets size with different p
Orange:Curve for components num for differnt p:
Text(0.5, 1.0, 'Curve for ER graph with different p')
```

Out[4]: