Слушателя учебной группы 7133 Байдина М.С.

Отчет по дисциплине «Теория графов»

Оглавление

[ПЗ от 28 Февраля 2](#_Toc35628912)

[ПЗ от 06 Марта 2](#_Toc35628913)

[ПЗ от 13 Марта 3](#_Toc35628914)

[ПЗ от 20 Марта 3](#_Toc35628915)

## ПЗ от 28 Февраля

На вход: 2 excel файла: “Дружба”, “Дни Рождения”, в каждом из которых находятся матрицы смежности.

Оценить, будут ли графы связными.

Для этого, сначала необходимо считать матрицу из файла:

file = open('Дружба.csv', 'r')  
f = open('list\_дружба', 'w')  
for (i, line) in enumerate(file):  
 for(j, bl) in enumerate(line[:-1].split(';')):  
 if bl == '1':  
 f.write(str(i) + '\t' + str(j) + '\n')  
file.close()  
f.close()

Далее, необходимо сравнить среднюю степень вершины с ln(n), где n – количество вершин в графе. Если средняя степень вершины > ln(n), то граф связен(может иметь несколько компонент связности, одна из которых содержит почти все вершины графа, остальные же содержат по две-три вершины).

!!!!ВСТАВИТЬ ПРАКИЧЕСКОЕ ПРИМЕНЕНИЕ ЭТОГО УТВЕРЖДЕНИЯ

## ПЗ от 06 Марта

Построить графическое представление компонент связности графа, представленного с помощью списка смежности.

import networkx  
import matplotlib.pyplot as plt  
g = networkx.readwrite.read\_edgelist(r"\*путь\_к\_файлу\_со\_списком смежности\*", comments='#', delimiter='\t')  
networkx.drawing.nx\_pylab.draw\_networkx(g)  
plt.show()

!!!!! ВСТАВИТЬ РЕЗУЛАТ ПРОГРАММЫ(КАРТИНКУ)

## ПЗ от 13 Марта

Задача: построить граф, с заданной вероятностью для связности

import numpy as np  
import random  
import networkx  
import matplotlib.pyplot as plt  
  
edjes = []  
count = 10  
#p=0.5\*(1/count)  
#p=1/count  
#p=0.5\*(1/count+np.log(count)/(count-1))  
p=2\*np.log((count)/(count-1))  
  
for i in range(count):  
 for j in range(i+1,count):  
 x=random.random()  
 if x < p:  
 edjes.append([i,j])  
 edjes.append([j,i])  
g = networkx.Graph()  
g.add\_edges\_from(edjes)  
networkx.drawing.nx\_pylab.draw\_networkx(g)  
plt.show()

!!!!! ВСТАВИТЬ РЕЗУЛАТ ПРОГРАММЫ(КАРТИНКУ)

Вывод: чем больше значение p, тем меньше компонент связности;

При p > np.log((count)/(count-1)) одна компонента связности.

## ПЗ от 20 Марта

Коэффициент кластеризации

- это характеристика сети, которая отображает степень взаимодействия между собой ближайших соседей данного узла.

Коэффициент кластеризации данного узла есть вероятность того, что два ближайших соседа этого узла сами есть ближайшие соседи.

Coef\_clust(i) = , где – реальное число связей, – степень узла.