

МИНОБРНАУКИ РОССИИ

ВЛАДИВОСТОКСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ  
(ФГБОУ ВО «ВВГУ»)

ИНСТИТУТ ЦИФРОВОЙ ЭКОНОМИКИ И БИЗНЕС-АНАЛИТИКИ

КАФЕДРА МАТЕМАТИКИ И МОДЕЛИРОВАНИЯ

Методические рекомендации по выполнению лабораторных работ по дисциплине

**ДИСКРЕТНАЯ МАТЕМАТИКА**

Часть 2(семестр 3)

Направление подготовки и направленность (профиль):

- 09.03.02 Информационные системы и технологии - Информационные системы и технологии
- 09.03.03 Прикладная информатика - Мобильные приложения и интеллектуальный анализ данных
- 09.03.04 Программная инженерия - Программная инженерия
- 10.05.03 Информационная безопасность автоматизированных систем - Безопасность открытых информационных систем
- 11.03.02 Инфокоммуникационные технологии и системы связи - Интернет-вещей и оптические системы и сети

Форма обучения:

очная

*Гренкин Г.В., кандидат физико-математических наук, доцент кафедры математики и моделирования,  
Gleb.Grenkin@vvsu.ru*

*Клочкова О.И., кандидат физико-математических наук, доцент, доцент кафедры математики и моделирования, [Klochkova.O@vvsu.ru](mailto:Klochkova.O@vvsu.ru)*

*Солодухин К.С., доктор экономических наук, профессор кафедры математики и моделирования,  
Konstantin.Solodukhin@vvsu.ru*

Утверждена на заседании кафедры математики и моделирования от \_\_\_\_\_, протокол № \_\_\_\_\_

## Лабораторная работа 2.1 Характеристики графа.

### Тема 2.1 Понятие графа и виды графов

#### Цели лабораторной работы

Сформировать у обучающихся умения и навыки построения графа и таблицу смежности в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

#### Планируемые результаты обучения

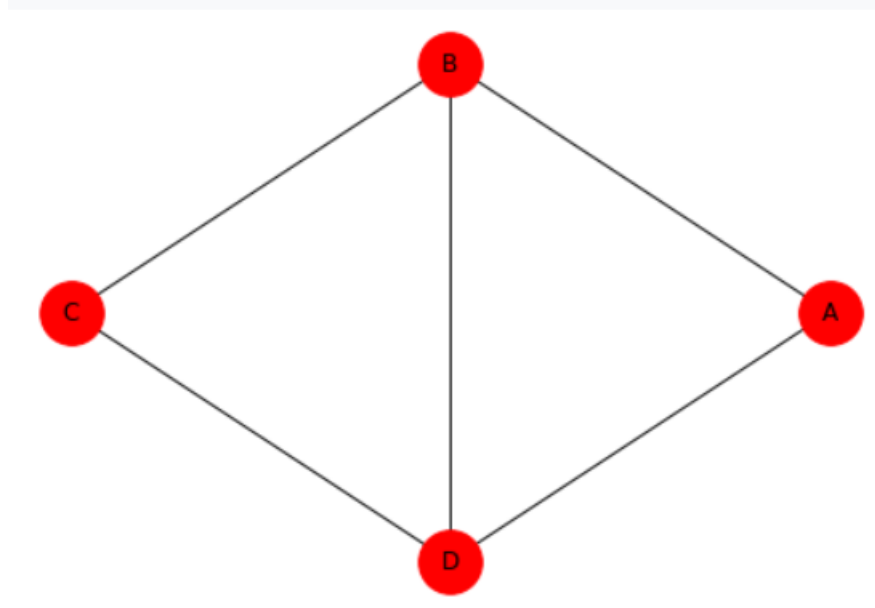
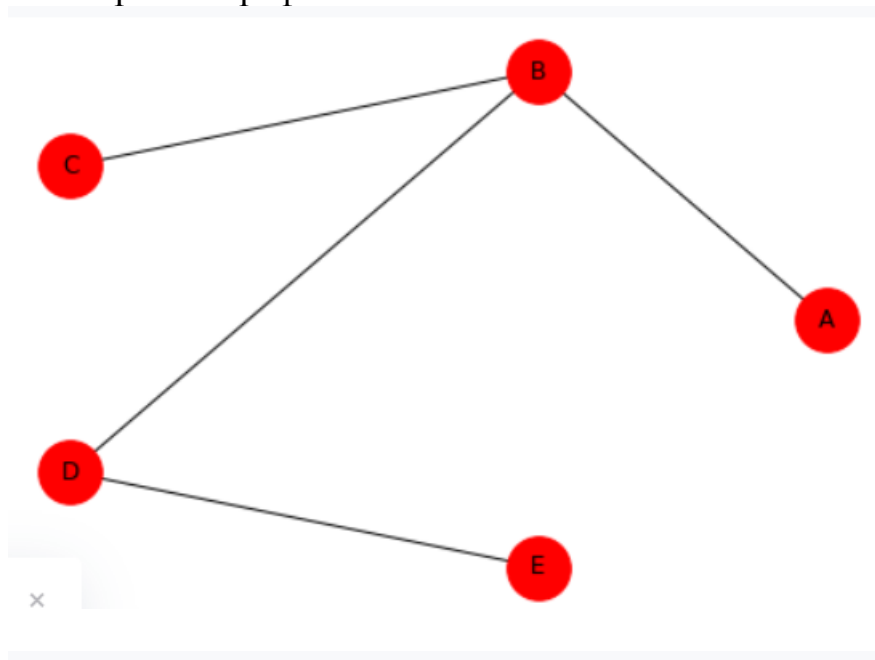
Формирование умения построить изображение заданного графа и таблицу смежности для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

#### Содержание лабораторной работы

#### Построить граф

#### Задание 1

#### 1. Построить графы



Возможно, в помощь ссылка <https://shwanoff.ru/networkx-part1/>

### Задание 2

Между населенными пунктами А,В,С,Д,Е построены дороги. Нужно определить длину кратчайшего пути между пунктами А и Е. Передвигаться можно только по дорогам, протяженность которых указана в таблице.

Задать расстояния между вершинами графа А, В,С,Д,Е с помощью таблицы смежности – определить длину кратчайшего пути между А и Е

Таблица расстояний графа

	А	В	С	Д	Е
А		1			
В	1		2	2	7
С		2			3
Д		2			4
Е		7	3	4	

Создать программу на Python, которая бы позволила

а) по вышеприведенной таблице **ввести** вершины графа

в)**вывести** расстояние между введенными вершинами

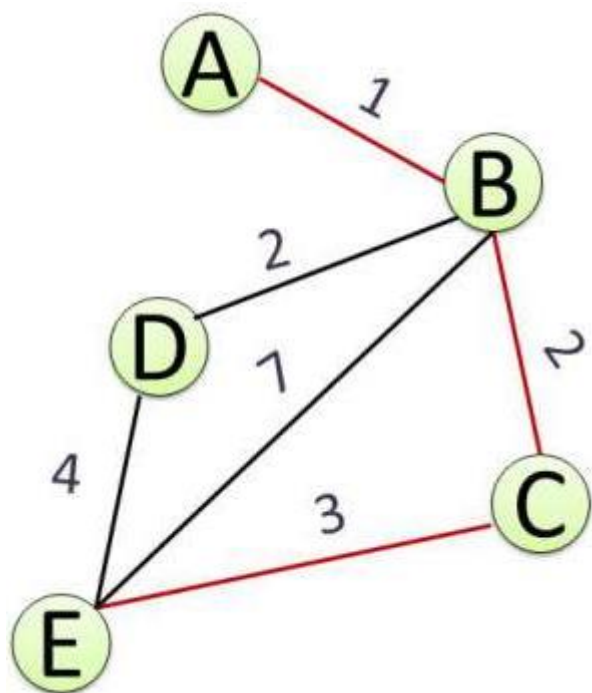
Тест

Вершины А,Е

Вывести результат Результат  $1+2+3= 6$

### Задание 3

Построить таблицу смежности графа (см. ниже)



- Создать программу на Python, которая бы позволила
- ввести новые расстояния между вершинами графа
  - по таблице **ввести** вершины графа
  - вывести** расстояние между введенными вершинами

#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с характеристиками графа, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

#### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

### Лабораторная работа №2.2 Визуализация графов. Тема 2.1 Понятие графа и виды графов

Работа с пакетом NetworkX и визуализацией графов с помощью пакета Matplotlib

- Введение.** Установить(если отсутствуют) пакеты NetworkX и Matplotlib

(При затруднениях попробуйте через свой смартфон в НАУШНИКАХ послушать и посмотреть по ссылке:

<https://www.youtube.com/watch?v=BXYsYLYGKEc>

<https://docs.microsoft.com/ru-ru/visualstudio/python/tutorial-working-with-python-in-visual-studio-step-05-installing-packages?view=vs-2022>

[https://www.youtube.com/watch?v=Qa\\_MysBUPbl](https://www.youtube.com/watch?v=Qa_MysBUPbl))

2. Ознакомьтесь с документацией по пакету NetworkX и примерами на странице <https://networkx.org/documentation/stable/tutorial.html>

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки визуализировать заданный граф в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

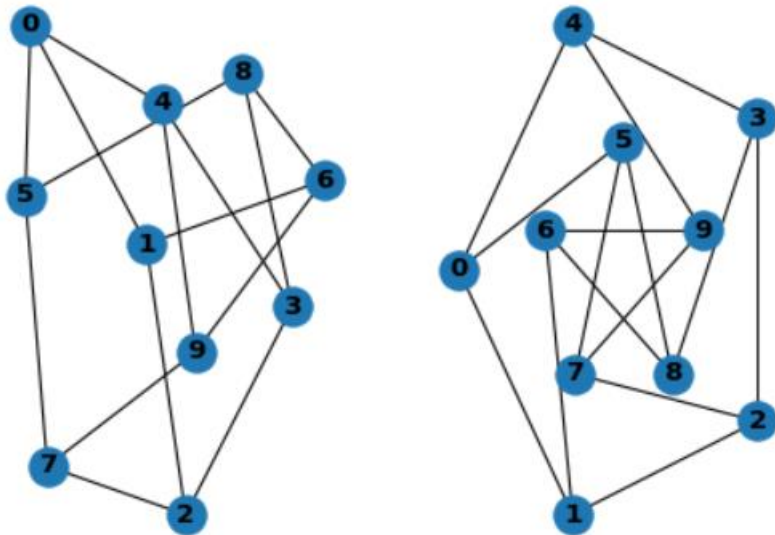
### Планируемые результаты обучения

Формирование умения визуализировать граф и определить таблицу смежности и число связности заданного графа для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

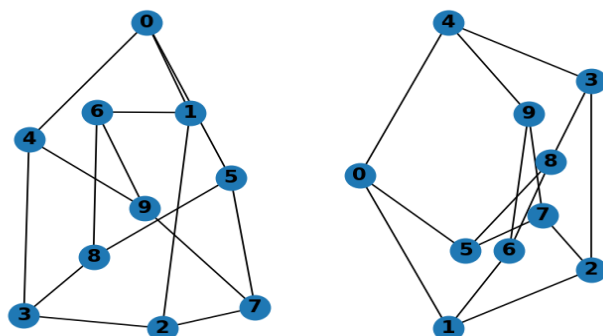
#### Задание

1. 3 Воспользуйтесь кодом на этой странице и организуйте вывод графов

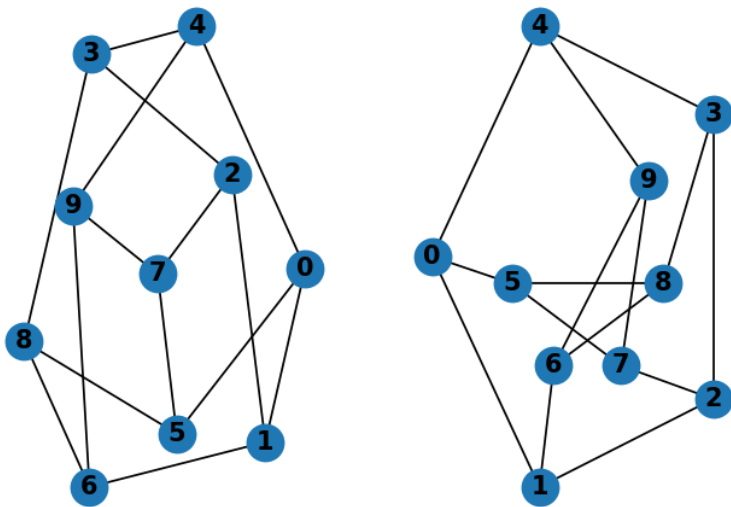


#### Задание 2

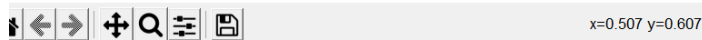
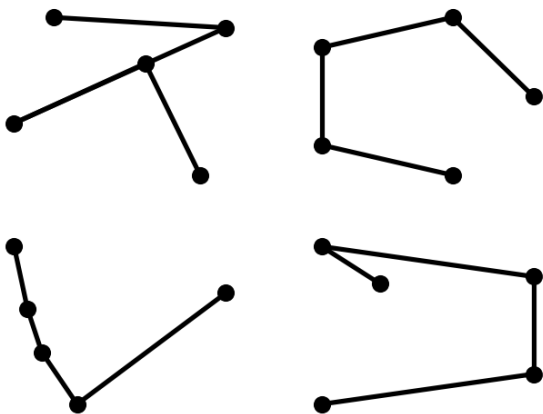
- 2.1 Измените код (минимально), чтобы получились графы



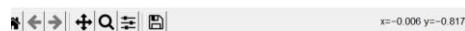
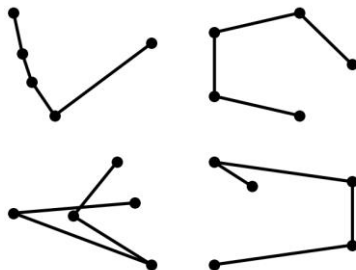
2.2. Измените код (минимально), чтобы получились графы



**Задание 3** Воспользуйтесь кодом на странице для получения графов

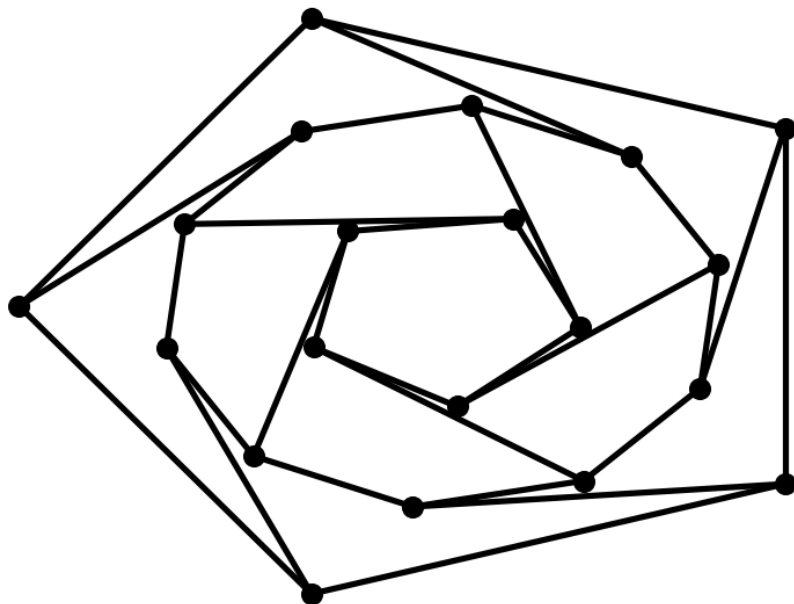


3.1 В какой строке кода задаются фигуры? Попробуйте поменять первую и третью графы местами



3.2 Обратите внимание на значения X и Y в правом нижнем углу. Координаты какой вершины они показывают?

**Задание 4** Измените код. Получите граф



x=-0.224 y=0.253

2). Координаты какой точки показаны?

Результаты показать преподавателю.

**Задание.** 5 По матрице смежности изобразить граф

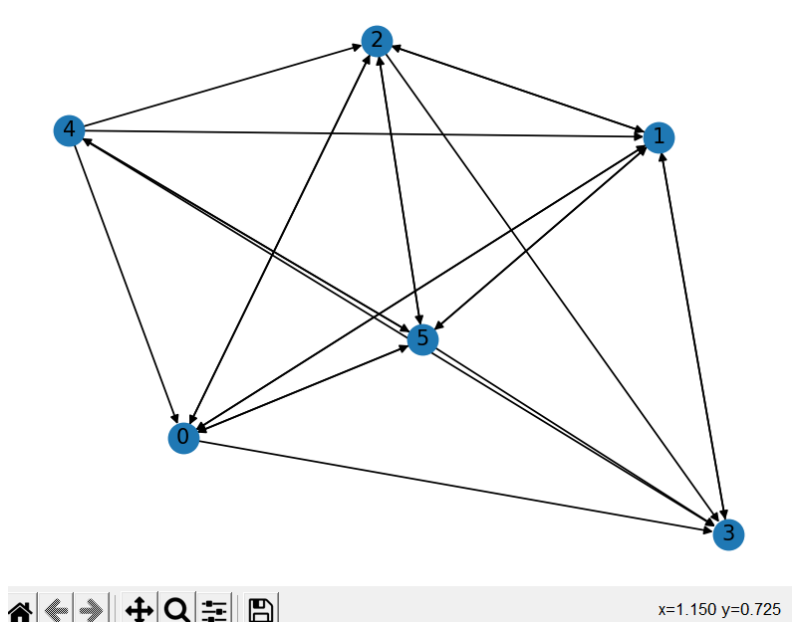
```
import networkx as nx
import numpy as np
import matplotlib.pyplot as plt
P0= [[0, 1, 1, 1, 0, 1],
      [1, 0, 1, 1, 0, 1],
      [1, 1, 0, 1, 0, 1],
      [0, 1, 0, 0, 0, 0],
      [1, 1, 1, 1, 0, 1],
      [1, 1, 1, 1, 1, 0]]

G = nx.DiGraph(np.matrix(P0))
nx.draw(G, with_labels=True, node_size=300, arrows=True)
plt.show()
```

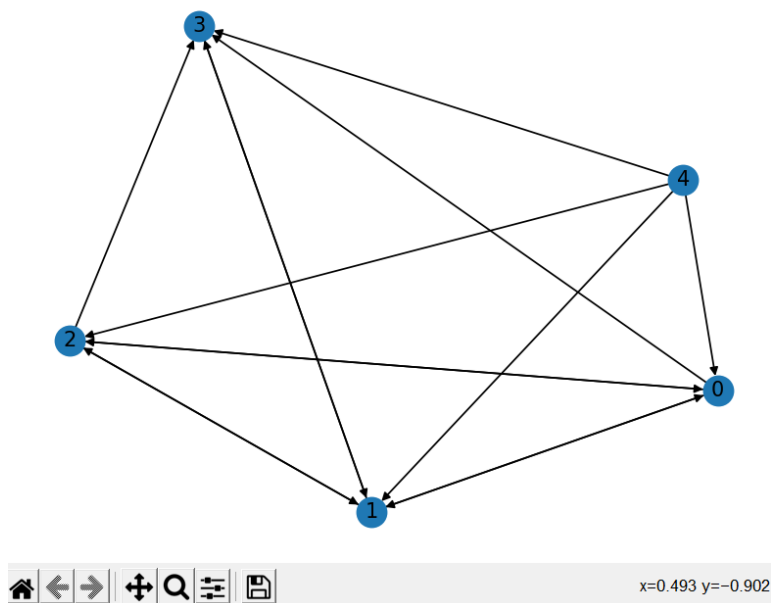
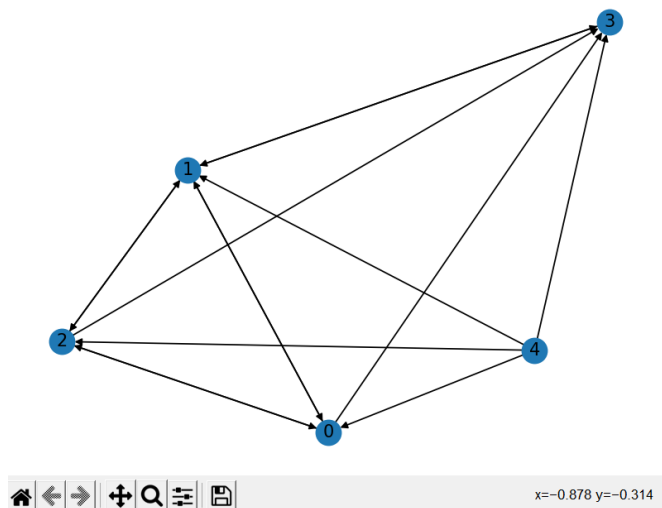
Воспользуйтесь примером по ссылке

<https://ru.stackoverflow.com/questions/1416356/%D0%9A%D0%B0%D0%BA-%D0%BD%D0%B0-python-%D0%BD%D0%B0%D0%BF%D0%B8%D1%81%D0%B0%D1%82%D1%8C-%D0%BA%D0%BE%D0%B4-%D0%B4%D0%BB%D1%8F-%D0%BF%D0%BE%D1%81%D1%82%D1%80%D0%BE%D0%B5%D0%BD%D0%B8%D1%8F-%D0%B3%D1%80%D0%B0%D1%84%D0%B0-%D0%BF%D0%BE-%D0%BC%D0%B0%D1%82%D1%80%D0%B8%D1%86%D0%B5-%D1%81%D0%BC%D0%B5%D0%B6%D0%BD%D0%BE%D1%81%D1%82%D0%B8-%D1%87%D1%82%D0%BE%D0%B1%D1%8B-%D1%82%D0%BE%D0%BB%D1%8C%D0%BA>





5.2 Измените код в матрице смежности, чтобы граф имел вид



**Задание 6** Организуйте ввод матрицы смежности с клавиатуры

По заданной таблице смежности или заданному изображению графа найти число связности графа и вывести изображения компонент связности

Требуется:

6.1.Изменить программу на языке программирования Python для вывода данных согласно заданной задаче

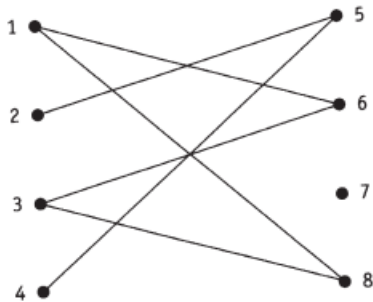
6.2.Вывести :

1)число связности

2)изображения компонент связности

ТЕСТОВЫЕ ДАННЫЕ

Граф имеет вид

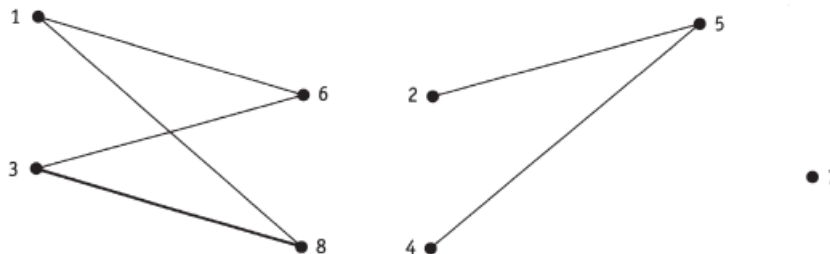


Результат

Решение (см. таблицу)

	$V'$	$c$
Исходные значения	{1, 2, 3, 4, 5, 6, 7, 8}	0
Выбор $y = 1$	{2, 4, 5, 7}	1
Выбор $y = 2$	{7}	2
Выбор $y = 7$	$\emptyset$	3

Таким образом,  $c(G)=3$ . Соответствующие компоненты связности имеют вид



*Краткие методические указания*

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных определением связности графа, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.

2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.3 Работа с геоданными Тема 2.2 Теория неориентированных графов

**Введение.** Определение расстояний на поверхности Земли

Размеры и форма Земли. Форма Земли отличается от шара и имеет несколько сплюснутую форму, близкую к сфероиду (эллипсоиду вращения), но истинная фигура Земли отличается и от сфероида, и от трехосного эллипсоида и не может быть представлена ни одной из известных математических фигур. Поэтому, говоря о фигуре Земли, имеют в виду не физическую форму земной поверхности, с океанами и материками, с их возвышенностями и впадинами, а так называемую поверхность геоида. Поверхность, нормальными к которой в любой из ее точек являются отвесные линии, называется уровенной поверхностью, или поверхностью равновесия. Уровенных поверхностей, как внутри Земли, так и охватывающих земную поверхность, или пересекающихся с ней, можно провести бесчисленное множество.

Та поверхность равновесия, которая совпадает в открытом океане с поверхностью покоящейся свободной воды, называется геоидом.

Для решения многих задач навигации и составления карт мелкого масштаба Землю принимают за сферу (шар).

Положение точки на земной сфере определяется сферическими координатами: сферической широтой и сферической долготой (в картографии применяют термин "географические координаты").

Сферическая широта точки  $A$  — угол  $\varphi_A$  между плоскостью экватора и направлением  $R$  на данную точку из центра земной сферы.

Сферическая долгота точки  $A$  — угол  $\lambda_A$ , заключенный между плоскостью нулевого (Гринвичского) меридиана и плоскостью меридиана данной точки.

Средний радиус Земли  $R = 6371210$  м.

Экваториальный радиус Земли  $R_{\text{Э}} = 6378,245$  м.

Полярный радиус Земли  $R_{\text{П}} = 6356,830$  м.

Длина дуги меридиана (дуги экватора, дуги окружности большого круга) в  $1^\circ$ ,  $1'$  и  $1''$  равна соответственно:

111 197 м (111,2 км), 1852 м (1,852 км) и 30,9 м



Кратчайшее расстояние между двумя точками на земной поверхности (если принять ее за сферу) определяется зависимостью:

$$\cos(d) = \sin(\varphi_A) \cdot \sin(\varphi_B) + \cos(\varphi_A) \cdot \cos(\varphi_B) \cdot \cos(\lambda_A - \lambda_B),$$

где  $\varphi_A$  и  $\varphi_B$  — широты,  $\lambda_A$ ,  $\lambda_B$  — долготы данных пунктов,  $d$  — расстояние между пунктами, измеряемое в радианах длиной дуги большого круга земного шара.

Расстояние между пунктами, измеряемое в километрах, определяется по формуле:

$$L = d \cdot R,$$

где  $R = 6371$  км — средний радиус земного шара.

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки перевода радианов в градусы и работы с гео-данными на в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения использовать методы неориентированных графов на гео-данных для последующего нахождения соответствующих организационно-управленческих решений. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1

1. Составить программу на языке Python для нахождения расстояний по заданным долготе и широте
2. Вывести расстояние между пунктами на Земле по заданным долготе и широте в градусах и радианах в виде таблицы (см.тест)

Тест. Найти расстояние между Тура (Россия) и Нью Йорком(США) и Тура и Сидней (Австралия)

	$\varphi$ (градус)	$\lambda$ (градус)	$\varphi$ (радиан)	$\lambda$ (радиан)	расстояние до Туры (км)
Тура Tura (Russia)	64,28 с.ш.	100,22 в.д.	1,121387	1,748224	—
Нью-Йорк (США) New-York (USA)	40,71 с.ш.	74,01 з.д.	0,710163	1,291063	8 335
Сидней (Австралия) Sydney (Australia)	33,874 ю.ш.	151,213 в.д.	0,590913	2,637827	11 780

#### Задание 2 XML и гео-данные (Владивосток)

2.1 Открыть карту гео-данных по адресу <https://www.openstreetmap.org/>

Найти Владивосток

2.2 Выбрать кнопку Экспортировать. Ограничиться данными, прилегающими к улице Гоголя

The screenshot shows the OpenStreetMap interface. At the top, there are buttons for 'Правка', 'История', and 'Экспорт'. Below the search bar, the 'Экспортировать' (Export) button is highlighted. The export options are displayed in a grid:

43.12266	
131.90365	131.90709
43.12008	

Below the export options, there is a 'Лицензия' (License) section with a blue 'Экспортировать' button. At the bottom, there is a note: 'Если этот экспорт пройдёт неудачно, пожалуйста, попробуйте один из перечисленных ниже источников:'.

Нажать кнопку Экспортировать. Появляется файл map1.osm

2.3 Скопировать файл в Вашу рабочую папку с файлами Python с новым именем map.xml- (у вас будет 2 файла: map.xml и map.

2.4 Откройте файл. Найдите в нем широту и долготу остановки «ВГУЭС» и «Некрасовская, 50»

```
<node id="844644521" visible="true" version="7"
  <tag k="bus" v="yes"/>
  <tag k="name" v="ВГУЭС"/>
  <tag k="public_transport" v="stop_position"/>
</node>
<node id="844644528" visible="true" version="10
  <tag k="bus" v="yes"/>
  <tag k="name" v="Некрасовская, 50"/>
```

Ориентируйтесь по тегу “node”

2.5 Посмотрите, какими тегами обозначается остановка автобуса и посчитайте общее количество остановок в экспортированной области.

Воспользуйтесь библиотекой BeautifulSoup из пакета bs4

2.6 Установите пакет lxml.

2.7 Код для подсчета остановок

```
1 from bs4 import BeautifulSoup
2 xml = open('map.osm', 'r', encoding='utf8').read()
3 soup = BeautifulSoup(xml, 'lxml')
4 print(soup)
5 cnt=0
6 for node in soup.find_all('node'):
7     for tag in node('tag'):
8         if tag['k'] == 'public_transport' and tag['v'] == 'stop_position':
9             cnt += 1
10 print(cnt)
11
```

Требуется вывести

2.8 Количество точек «Суши- бар»

2.9 Найти расстояния

между остановками

между ВГУЭС и ТГМУ

между ВГУЭС и ДВФУ

### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с неориентированными графами, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -обе части
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание

3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.4 Алгоритм поиска кратчайшего расстояния в глубину, ширину, Дейкстры. Темы 2.2 Теория неориентированных графов

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки построения кратчайшего расстояния заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения определить кратчайший маршрут заданного графа для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1 Поиск в глубину DFS -(0,5 балла)

- 1.1 Ознакомьтесь с документацией на странице <https://habr.com/ru/company/otus/blog/660725/>
- 1.2 С помощью приведенного кода организуйте поиск в глубину для приведенного на упомянутой странице примера. Результат показать преподавателю

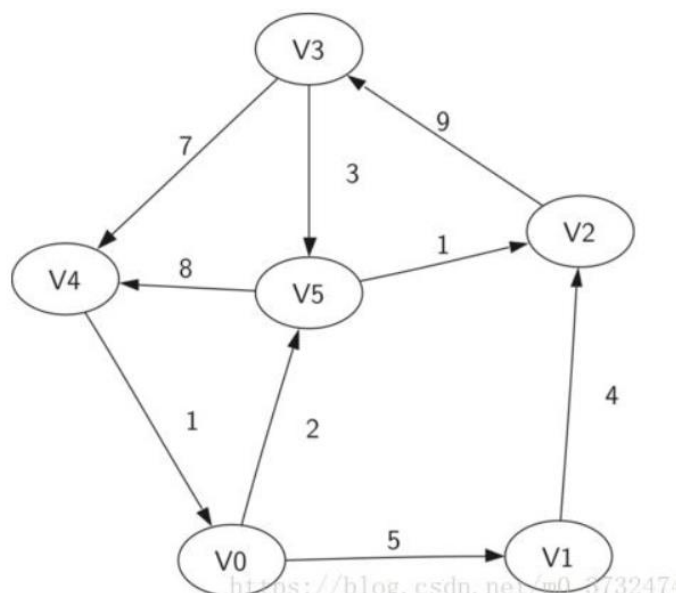
#### Задание 2 Поиск в ширину BFS-(0,5 балла).

- 2.1 Ознакомьтесь с документацией <https://russianblogs.com/article/2700458711/>
- 2.2 Воспользуйтесь кодом на этой странице и организуйте ввод ориентированного графа

$V=\{V0,V1,V2,V3,V4,V5\};$

Набор из 9 ребер:

$E=\{(v0,v1,5),(v1,v2,4),(v2,v3,9),(v3,v4,7),(v4,v0,1),(v0,v5,2),(v5,v4,8),(v3,v5,3),(v5,v2,1)\}$



2.3 Воспользуйтесь кодом на этой странице и выведите ребра для изображенного графа

```
( 0 , 1 )  
( 0 , 5 )  
( 1 , 2 )  
( 2 , 3 )  
( 3 , 4 )  
( 3 , 5 )  
( 4 , 0 )  
( 5 , 4 )  
( 5 , 2 )
```

**Задание 3**(2 балла)

Создать программу на Python, которая бы позволила

а) ввести матрицу смежности взвешенного графа

б) вывести результат минимальный путь из вершины V0 в V3 и обратно с помощью алгоритма поиска в глубину или ширину

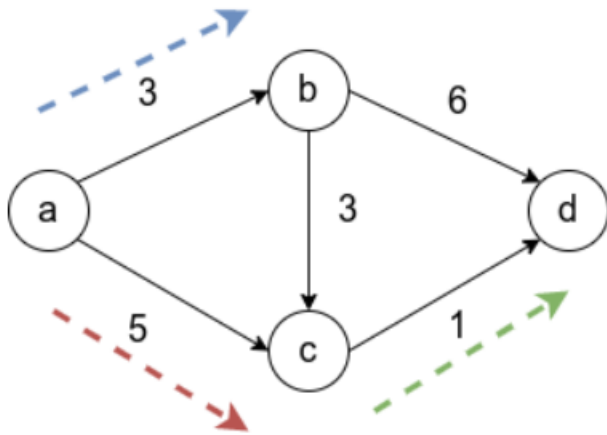
	V0	V1	V2	V3	V4	V5
V0		5				2
V1			4			
V2				9		
V3					7	3
V4	1					
V5			1		8	

Результат показать преподавателю

**Задание 4.** Алгоритм Дейкстры.

4.1 Ознакомьтесь с информацией на странице <https://habr.com/ru/company/otus/blog/599621/>

4.2 Пользуясь приведенным кодом организуйте алгоритм Дейкстры для графа



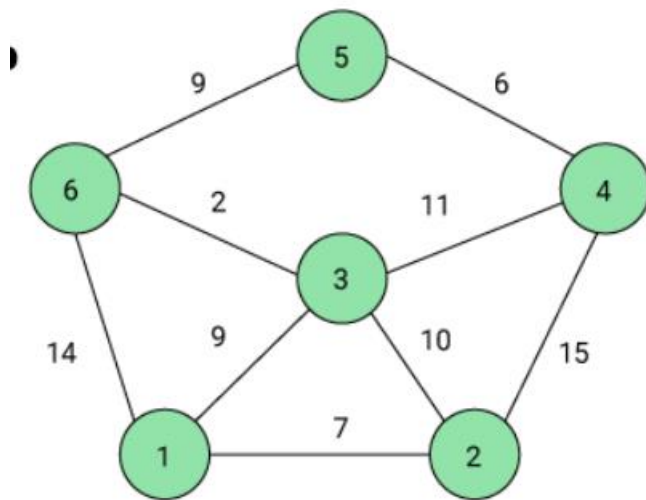
4.3 В качестве теста определите кратчайшее расстояние ас, ad

4.4 Для графа, изображенного ниже: (2 балла)

создать программу на Python, которая бы позволила

а) ввести матрицу смежности взвешенного графа

б) вывести результат определить кратчайшее расстояние между вершинами 1-6, 1-5, 1-4 по алгоритму Дейкстры



Результат показать преподавателю

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка



	решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.5 Алгоритм поиска Дейкстры на геоанных. Тема 2.2 Теория неориентированных графов

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки построения кратчайшего расстояния заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения определить кратчайший маршрут заданного графа для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1

1. Ознакомиться с информацией на странице

<https://waksoft.susu.ru/2021/10/21/realizacziya-algoritma-dejkstry-na-python/>

Требуется:

- 1.Использовать код на странице
- 2.Вывести :
  - 2.1Путь (ценность) маршрута из Рейкьявика в Белград
  - 2.2Наиболее короткий маршрут из Рейкьявика в Белград
  - 2.3. Пересчитать маршрут в км, используя широту и долготу городов из Рейкьявика в Белград на карте

#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с особенностями построения кратчайшего расстояния в графах

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

#### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.

2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.6 Деревья. Код Прюфера по дереву. Тема 2.3 Эквивалентные определения деревьев

**Введение.** Граф  $G(V, E)$  называется *деревом*, если он связан и ациклический, т.е. не содержит циклов. Остовное дерево строится просто: выбирается любое ребро и последовательно добавляются другие ребра, не создавая при этом циклов. Остовное дерево с наименьшим весом называется минимальным остовным деревом (МОД). По дереву строится код Прюфера (см. лекции Райгородского А.М. на платформе открытого образования [openedu.ru](https://openedu.ru))

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки построения кода Прюфера из заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения построения кода Прюфера из заданного графа и дерева по заданному коду Прюфера для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1

Воспользуйтесь кодом на странице по ссылке и напишите программу для декодирования кода Прюфера:

Ввод кода Прюфера и номера вершин

Вывод ребер дерева

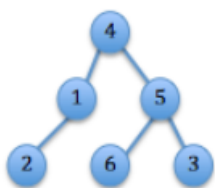
Тест

Ввод: Код P: 1 4 5 5

Вершины V : 1 2 3 4 5 6

Вывод ребер 21 14 65 35 54

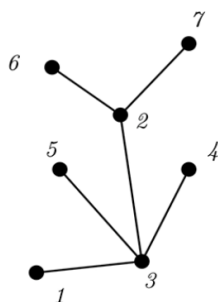
Дополнительный балл за визуализацию дерева



#### Задание 2.

Напишите программу для построения кода Прюфера заданному дереву.

#### Тестирование



Ввод вершин 1 2 3 4 5 6 7

Ввод ребер 13 34 23 27 26 35

Вывод: Код Прюфера 33322

Показать преподавателю

*Краткие методические указания*

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с построением остова дерева, изложенных на лекционных занятиях Райгородского А.М. на платформе открытого образования [openedu.ru](http://openedu.ru))

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## **Лабораторная работа №2.7 Деревья. Применение деревьев в шифровании. Тема 2.3 Эквивалентные определения деревьев**

**Введение.** В шифровании используется помимо азбуки Морзе используется операция XOR с ключом

### **Цели лабораторной работы**

Сформировать у обучающихся умения и навыки шифрования – дешифрования по ключу и использования деревьев в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### **Планируемые результаты обучения**

Формирование умения построения шифрования – дешифрования слова по ключу и по дереву для последующего применения при решении практических задач. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### **Содержание лабораторной работы**

#### **Задание 1**

1.1 Ознакомьтесь с информацией на странице

<http://nikulux.ru/python-praktika/konverter-russkogo-teksta-v-azbuku-morze-na-python-3/>

1.2 Измените код, чтобы была возможность ввода слова. Выключите звук Покажите преподавателю

## Задание 2

2.1 Замените точки на нули и тире на единицы.

Покажите результат преподавателю

2.2 Шифрование.

2.2.1 Введите модуль шифрования на основе ключа (пример)

**Применение XOR: шифрование-дешифрование с одним и тем же ключом**

$(x \oplus y) \oplus y = x$

**Data:** M O S C O W  
ASCII Код: 1001101 1001111 1010011 1000011 1001111 1010111

Для шифрования используем ключ – конкретную строку (из Библии)

**Key:** C O M P U T  
ASCII Код: 1000011 1001111 1001101 1010000 1010101 1010100

Получаем шифровку = Data  $\oplus$  Key

**Шифровка :** 0001110 0000000 0011110 0010011 0011010 0000011

Шифровку передаем Штирлицу. Он знает, какая строка из Библии - ключ.  
Для дешифрования он использует ту же строку: Шифровка  $\oplus$  Key = Data

**Data:** 1001101 1001111 1010011 1000011 1001111 1010111

2.2.2 В качестве ключа возьмите слово ВГУЭСВВГУ

2.2.3 В качестве теста выведите слово ПОБЕДА

Результат показать преподавателю

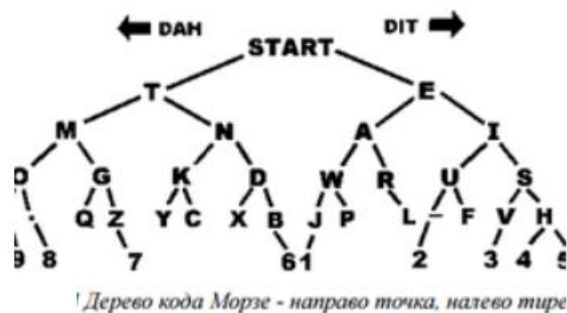
## Задание 3 Шифрование с помощью алгоритма Хоффмана

3.1 Ознакомьтесь с документацией по алгоритму Хаффмана

<https://practice.keyfire.ru/blog/kak-zakodirovat-stroku-s-pomoshyu-algoritma-haffmana/>

3.2 Воспользуйтесь кодом на этой странице и закодируйте слова 'abba', SOS затем раскодируйте эти слова

## Задание 4 Шифрование с помощью дерева



4.1 Измените кодирование букв согласно дереву. Попробуйте закодировать буквы латинского алфавита

4.2 В качестве теста передайте слово SOS

Результат показать преподавателю

#### *Краткие методические указания*

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с построением остова дерева, изложенных на лекционных занятиях Райгородского А.М. на платформе открытого образования [openedu.ru](https://openedu.ru))

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

#### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.8 . Гамильтоновы циклы. Циклы Эйлера

### Тема 2.4 Формула для числа унициклических графов

**Введение.** Уциклическим называется граф, состоящий из единственного цикла, или, другими словами, некоторого числа вершин, соединённых замкнутой цепью. Граф-цикл с  $n$  вершинами обозначают как  $C_n$ . Число вершин в  $C_n$  равно числу рёбер и каждая вершина имеет степень 2, то есть любая вершина инцидентна ровно двум рёбрам. В отличие от *эйлеровых графов*, обладающих замкнутым маршрутом, который проходит по всем ребрам графа ровно один раз, *гамильтонов цикл* проходит через каждую вершину в точности один раз. Граф с таким циклом называется *гамильтоновым графом*. Важность задачи нахождения гамильтонова цикла связана с решением графической модели задачи коммивояжера. Эта модель состоит из гамильтонова графа, вершины которого представляют города, а ребра- связывающие их дороги. Каждое ребро снабжено весом, связанным с транспортными затратами. Для решения задачи необходимо найти гамильтонов цикл с минимальным общим весом.

#### Цели лабораторной работы

Сформировать у обучающихся умения и навыки нахождения гамильтоновых циклов с наименьшим весом заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

#### Планируемые результаты обучения

Формирование умения определить гамильтоновы циклы с наименьшим весом заданного графа для последующего применения при решении практических задач коммивояжера. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

#### Содержание лабораторной работы

##### *Задание 1*

Воспользуйтесь информацией семинара 5 цикла Райгородского А.М.

Напишите код на языке Python для расчета количества гамильтоновых циклов для полного графа  $K_n$

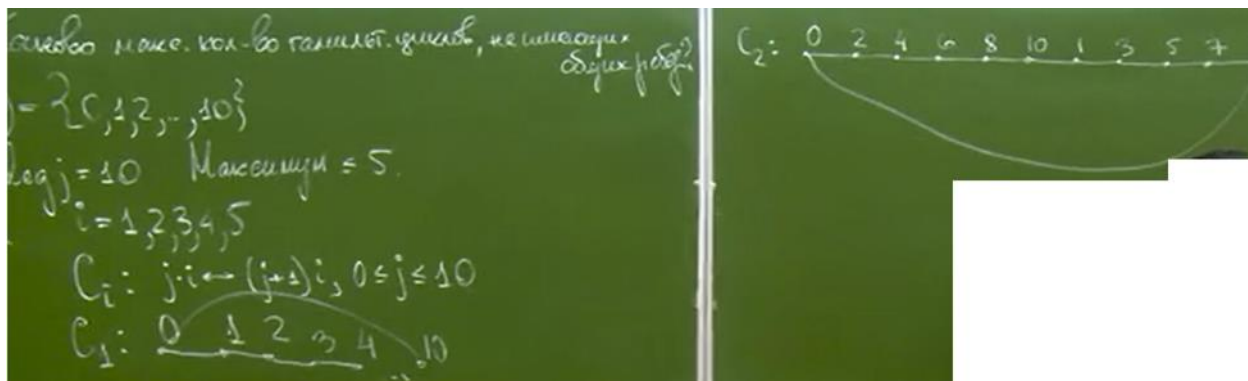
Ввод значения  $n$

Вывод максимального количества гамильтоновых циклов, не имеющих общих ребер

Тест

Ввод:  $n=11$

Вывод 5 циклов – перечислить вершины, входящие в цикл: первые 2 цикла на рисунке



## Задание 2

Визуализировать полученное дерева

### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных семинара 5 и к нему лекций Райгородского А.М. на платформе открытого образования orepedu.ru вопросов, связанных определением гамильтоновых циклов графа .

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.9 . Полные и планарные графы

### Тема 2.5 Планарность и критерий Куратовского

**Введение.** Граф называется полным, если любые две его вершины соединены ребром.

Обозначают:  $K_n$  . Граф является планарным тогда и только тогда, когда он не содержит подграфов, гомеоморфных  $K_5$  и  $K_{3,3}$ .

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки нахождения гамильтоновых циклов с наименьшим весом заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения определить планарность заданного графа для последующего применения при решении практических задач логистики. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1 (1 балл)

Создать программу на Python, которая бы позволила

- Организовать ввод количества вершин для полного графа и дерева
- вывести результат -количество ребер

#### Задание 2 (2 балла)

Организовать ввод степенной последовательности

Вывести возможен ли такой граф. В случае невозможности вывести причину (см.лекцию в Moodle)

#### Задание 4 (2 балла)

Организовать ввод количества вершин для пустого графа, полного графа, дерева и простого цикла  $C_n$

Вывести хроматическое число

#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных семинара 5 и к нему лекций Райгородского А.М. на платформе открытого образования [openeu.ru](http://openeu.ru) вопросов, связанных определением гамильтоновых циклов графа .

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

#### Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

### Лабораторная работа №2.10.Паросочетания. Венгерский алгоритм.

#### Тема 2.6 Паросочетания. Теорема Холла и Кенига

**Введение.** Появление венгерского метода (венгерского алгоритма) связано с решением задачи о назначениях.

Суть: имеется 5 работ (столбцы) и столько же работников (строки), которые имеют разную оплату за работы (см матрицу) Нужно так распределить работников по работам, чтобы суммарная оплата была минимальной:

	1	2	3	4	5
A	32	28	4	26	4
B	17	19	4	17	4
C	4	4	5	4	4
D	17	14	4	14	4
E	21	16	4	13	4

Если выбирать минимальные значения наобум, с учетом того, что каждый работник может занять только одну работу, можно получить (начиная с пятой строки) общую суммарную зарплату  $4+4+4+17+28= 57$ .

Венгерский алгоритм позволяет уменьшить это число до 39.

Определенным способом сводят матрицу, приведенную выше к матрице, где в каждой строке будет нулевое значение. Позиция нулей и указывает – какие цифры в вышеприведенной матрице взять

18	14	0	12	0
3	5	0	3	0
0	0	11	0	10
3	0	0	0	0
8	3	1	0	1

:

из пятой строки нужно взять предпоследний элемент 13

из четвертой 14, из третьей первый элемент 4 и два минимальных элемента из первой и второй строки (по 4). Итого  $13+14+4+4+4= 39$

**Венгерский алгоритм.** Програмируя венгерский алгоритм, нужно выделить 5 шагов

Шаг 1. Вычтите минимальный элемент в каждой строке из своей строки (для всех строк)

Шаг 2. Вычтите минимальный элемент в каждом столбце из своего столбца (для всех столбцов)

Шаг 3. Используйте наименьшее количество горизонтальных или вертикальных линий, чтобы покрыть все нули, например,

15	0	0	0
0	50	20	25
35	5	0	10
0	65	50	65

Шаг 4. Поскольку общее количество горизонтальных и вертикальных линий равно 3, что меньше 4, введите Шаг 5

Шаг 5 Минимальное значение, которое не покрывается, равно 20, минимальное значение 20 вычитается из каждой строки, которая не покрывается, и минимальное значение 20 добавляется к каждому покрытому столбцу, а затем переходите к шагу 3.



$$\begin{bmatrix} 20 & 0 & 5 & 0 \\ 0 & 45 & 20 & 20 \\ 35 & 0 & 0 & 5 \\ 0 & 60 & 50 & 60 \end{bmatrix} \rightarrow \begin{bmatrix} 20 & 0 & 5 & 0 \\ -20 & 25 & 0 & 0 \\ 35 & 0 & 0 & 5 \\ -20 & 40 & 30 & 40 \end{bmatrix} \rightarrow \begin{bmatrix} 40 & 0 & 5 & 0 \\ 0 & 25 & 0 & 0 \\ 55 & 0 & 0 & 5 \\ 0 & 40 & 30 & 40 \end{bmatrix}$$

Шаг 3. Используйте наименьшее количество горизонтальных или вертикальных линий, чтобы покрыть все нули

$$\begin{bmatrix} 40 & 0 & 5 & 0 \\ 0 & 25 & 0 & 0 \\ 55 & 0 & 0 & 5 \\ 0 & 40 & 30 & 40 \end{bmatrix}$$

Шаг 4. Поскольку общее количество горизонтальных и вертикальных линий равно 4, алгоритм завершается, и результат распределения показан на рисунке ниже.

$$\begin{bmatrix} 40 & 0 & 5 & 0 \\ 0 & 25 & 0 & 0 \\ 55 & 0 & 0 & 5 \\ 0 & 40 & 30 & 40 \end{bmatrix} \rightarrow \begin{bmatrix} 90 & 75 & 75 & 80 \\ 35 & 85 & 55 & 65 \\ 125 & 95 & 90 & 105 \\ 45 & 110 & 95 & 115 \end{bmatrix}$$

В этом примере исходной матрицей была крайняя правая матрица. Нули левой матрицы указывают на адреса значений, которые нужно выбрать в исходной матрице. В данном случае минимальная суммарная оплата составляет  $45+95+55+80=275$ .

Код алгоритма на языке Python можно посмотреть по ссылке <https://russianblogs.com/article/5547740509/>

### Цели лабораторной работы

Сформировать у обучающихся умения и использования венгерского алгоритма для нахождения оптимального решения в рамках теоретических и практических исследований для последующего использования при решении задачи о назначениях. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения применения венгерского алгоритма для нахождения оптимального решения задачи и последующего применения при решении задач о назначениях. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание

1. По заданной таблице о назначениях найти оптимальное решение, применить венгерский алгоритм, распечатать суммарное значение заработной платы

Требуется:

1.Использовать программу на языке программирования Python для ввода данных согласно заданной задаче

2.Вывести :

1)минимальную стоимость и лучшие значения

2)изображения таблицы с промежуточным решением

ТЕСТОВЫЕ ДАННЫЕ

Начальная таблица

```
cost matrix =  
[[12 53 31 40 47]  
 [ 2 21 72 61 17]  
 [70 72 85 54 39]  
 [93 34 62 75 51]  
 [76 14 15  7 72]]
```

Результат работы алгоритма

```
min cost = 113  
best solution = [31, 2, 39, 34, 7]
```

## Задание 2 Венгерский алгоритм в задаче о назначениях

Ознакомьтесь с информацией по ссылке

<https://habr.com/ru/post/422009>

Примените код в предыдущем задании к рассмотренной задаче о назначениях

*Краткие методические указания*

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с применением венгерского алгоритма.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение конца таблицы
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.11. Алгоритмы из природы. Тема 2.7 Экстремальная теория графов

**Введение.** Одной из наиболее интенсивно разрабатываемых проблем оптимального управления остается задача коммивояжера и ее модификации. Использование экстремальной теории графов (метод переборных, метод Дейкстры и т.д.) с учетом применения критериев, основанных на степени вершин графов, оказались очень полезны. Ранее большое внимание уделяется задаче М коммивояжеров [1] —[6]. Алгоритм, заимствованный из природы, называется муравьиным алгоритмом (ACO – Ant Colony Optimization). Граф, на котором будет происходить поиск решения, можно сгенерировать случайным образом. [7].

## Цели лабораторной работы

Изучить Муравьиный алгоритм. Граф, на котором будет происходить поиск решения, можно сгенерировать случайным образом. Сначала генерируются координаты точек, затем при помощи модуля `spatial` из библиотеки `scipy` происходит вычисление матрицы расстояний (расстояние от каждой вершины до каждой вершины). заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

## Планируемые результаты обучения

Формирование умения воспользоваться алгоритмами, заимствованными из природы, для последующего применения при решении практических задач составления маршрутов движения по заданной схеме улиц. Использовать понятие случайного графа Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

## Содержание лабораторной работы

### Задание 1 Муравьиный алгоритм

1.1 Ознакомьтесь с информацией по ссылке [7]

<https://vc.ru/newtechaudit/353372-muravi-i-python-ishchem-samye-korotkie-puti>

1.2 На языке Python создать программу для ввода количества городов

1.3 Вывести количество путей для 5 городов согласно приведенной на странице формулы  
Показать преподавателю

### Задание 2 Метод перебора вариантов в муравьином алгоритме (4 балла)

2.1 Воспользуйтесь кодом, на странице [7] для реализации муравьиного алгоритма на языке Python

2.2 Организуйте ввод количества муравьёв, равным 40 (`size_pop=40`), итераций, равным 30 (`max_iter=30`) и вершин – 40.

2.3 Вывод длины пути.

2.4 Вывод графиков вывод графиков

Результаты показать преподавателю

#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных определением ориентированных графов, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

## Шкала оценки

Баллы	Описание
5	Выполнены оба задания полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение конца таблицы
4	Выполнены оба задания полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задания выполнены не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.12. Ориентированные графы. Тема 2.8 Ориентированные графы

**Введение.** Ребра ориентированных графов имеют направления. Для поиска кратчайшего расстояния можно использовать уже рассмотренные методы, например, алгоритм Дейкстры (см. лабораторную работу 5) так и другие методы, например - алгоритм Флойда Уоршелла с отрицательными весами.

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки нахождения циклов Терри заданного графа в рамках теоретических и практических исследований для последующего использования при решении практических задач. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

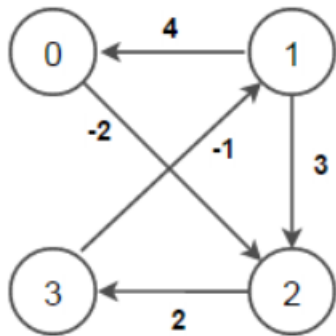
Формирование умения находить кратчайшее расстояние ориентированного графа с отрицательными весами с помощью - алгоритма Флойда Уоршелла для последующего применения при решении практических задач составления маршрутов движения по заданной схеме улиц. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1

1. Ознакомьтесь с документацией на странице <https://www.techiedelight.com/ru/pairs-shortest-paths-floyd-warshall-algorithm/>

С помощью приведенного кода организуйте вывод пути и оценку пути с учетом веса для графа



2. Сделать все веса положительными и организуйте вывод пути и оценку пути с учетом веса
3. Используя алгоритм Флойда, вывести пути и оценку пути с учетом веса для зеленого графа (часть 1 задание 5.4) и сравнить с результатами, полученными по алгоритму Дейкстры при определении кратчайшего расстояния между вершинами 16,15,14

Результаты показать преподавателю

#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных определением ориентированных графов, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

## Шкала оценки

Баллы	Описание
5	Задание выполнено полностью и абсолютно правильно. Тестовое задание выполнено -включая изображение конца таблицы
4	Задание выполнено полностью и правильно, но решение содержит некоторые неточности и несущественные ошибки или выполнено 1 тестовое задание, например, без изображения
3	Задание выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
2	Задание выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Лабораторная работа №2.13. Потoki в сетях. Алгоритм Форда -Фалкерсона Тема 2.9 Потoki в сетях

**Введение.** Ребра ориентированных графов имеют направления. Для поиска кратчайшего расстояния

### Цели лабораторной работы

Сформировать у обучающихся умения и навыки использования сетевых моделей в рамках теоретических и практических исследований для последующего принятия организационно-управленческих решений. Повысить навыки программирования на языке Python

### Планируемые результаты обучения

Формирование умения использовать сетевые модели для последующего нахождения соответствующих организационно-управленческих решений. Формирование навыков программирования и использования методов естественнонаучных дисциплин в профессиональной деятельности для теоретического и экспериментального исследования.

### Содержание лабораторной работы

#### Задание 1 Стоки - истоки

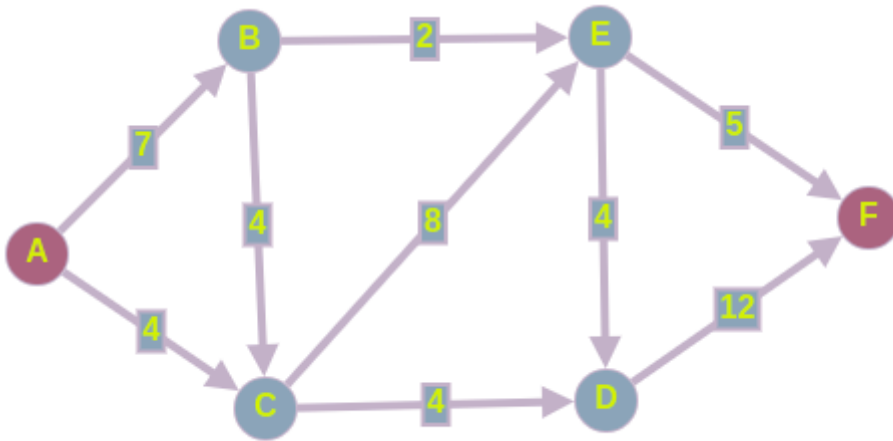
Ознакомьтесь с документацией на странице

<https://habr.com/ru/post/566248/>

Постановка задачи

Имеется следующий *ориентированный граф*, в котором вес ребра обозначает пропускную способность между вершинами. Нужно найти *максимальный поток*, который можно пропустить

из *истока* в *сток* .



Исходный граф

Как работает сам алгоритм?

Следует понимать *остаточную сеть* как тот же граф, который имеется на входе, но в этом случае мы будем производить над ним некоторые операции:

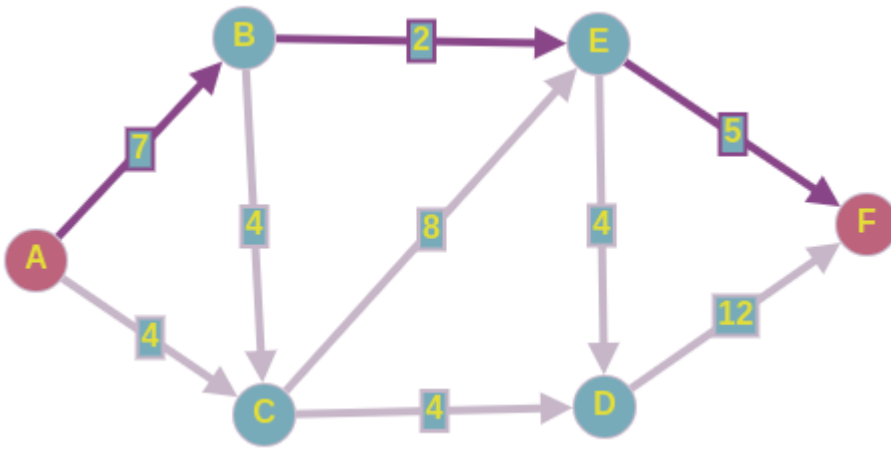
1. Отправлять определенное количество потока из текущей вершины в соседние.
2. Возвращать определенное количество потока из соседних вершин в текущую.
  - В начальный момент времени поток, который мы хотим провести через нашу сеть, должен быть равен нулю. *Остаточная сеть* совпадает с исходной сетью.
  - Находим **любой** путь из *истока* в *сток* в *остаточной сети*. Если путь не находим, утверждается, что поток является максимальным.
  - Пускаем через найденный путь поток равный минимальному весу ребра, которое входит в множество ребер найденного пути.
  - Из веса ребер на этом пути высчитываем размер потока, который мы пустили.
  - А к весу обратных ребер (будем считать, что они существуют в *остаточной сети* и равны 0) прибавляем размер потока. Другими словами, на предыдущем шаге мы отправили некоторое количество потока из текущей вершины в следующую, а теперь при желании можем вернуть это же количество потока обратно в текущую.
  - Возвращаемся обратно к нахождению пути в *остаточной сети* после модификации.

1. Требуется написать (найти) код алгоритма Форда -Фалкерсона на языке Python

2. Вывести максимальный поток

3. В качестве теста рассмотреть пример для ориентированного графа

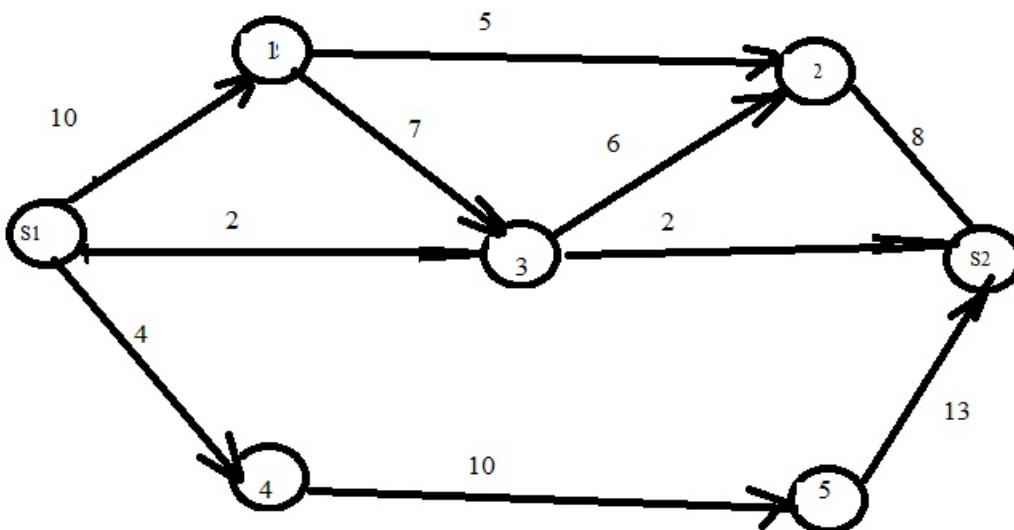
**Тест**



Максимальный поток в сети по данному графу 10 ед.

### Задание 2

2.1. Вычислить максимальный поток для графа на рисунке



#### Краткие методические указания

При подготовке к лабораторной работе необходимо обратить внимание на содержание основных теоретических вопросов, связанных с сетевыми моделями, изложенных на лекционных занятиях.

Отчет о выполнении лабораторной работы должен быть представлен в электронном виде (на языке программирования Python).

#### Шкала оценки

Баллы	Описание
4	Задание 1 и задание 2 выполнено полностью и абсолютно правильно.
3	Задание 1 выполнено полностью и правильно, задание 2 содержит некоторые неточности и несущественные ошибки.
2	Задание 1 выполнено не полностью, с существенными ошибками, но подход к решению, идея решения, метод правильны.
1	Задание 1 выполнено частично, имеет ошибки, осуществлена попытка решения на основе правильных методов и идей решения.
0	Задание не выполнено.

## Список использованной литературы

1. Svestka G. A., Huckfeldt V. E. A computational experience with an M-salesman traveling salesman problem // *Manag. Sci.* 1973. V. 19. № 7. P. 790-799.
2. Bellmore M.t Honig S. Transformation, o f multisalessman problem to the standart traveling salesman problem//*J. A.C. M.* 1974. V. 21. № 3. P. 500-504.
3. Rao M. R. A note on the multiple traveling salesman problem // *Operat. Res.* 1980. V. 28. № 3. Pt. 1. P. 628-632.
4. Husban A. An exact solution method for the MJSP//*J. Operat. Res. Soc.* 1989. V. 40. № 5. P. 461-469.
5. Laporte G., Nobert J. A cutting planes algorithm for the M-salesman problem// *J. Operat Res. Soc.* 1980. V. 31. № 11. P. 1017-1023.
6. Меламед И. И., Сергеев С. И., Сигал И. Х. Задача коммивояжера. Приближенные алгоритмы//*Автоматика и телемехан.* 1989. № И. С. 3-24
7. Муравьи и Python: ищем самые короткие пути [Электронный ресурс] // Osnova Платформа для контентных проектов. – Режим доступа: <https://vc.ru/newtechaudit/353372-muravi-i-python-ishchem-samyе-korotkie-puti> [дата обращения – 28.06.2023].