Оглавление

[Занятие 2 Проверено 2](#_Toc181199403)

[ЗАНЯТИЕ 3 Проверено: 7](#_Toc181199404)

[Занятие 4: Изоморфизм СДАНО 11](#_Toc181199405)

[Занятие 5 Операции над графами СДАНО 19](#_Toc181199406)

[Занятие 6. Код Прюфера СДАНО 22](#_Toc181199407)

[Занятие 7. Алгоритм поиска в глубину и поиска в ширину СДАНО 25](#_Toc181199408)

[Занятие 8 Алгоритм поиска минимального остовного дерева: СДАНО 27](#_Toc181199409)

[Занятие 9. Эйлеровы и гамильтоновы графы 28](#_Toc181199410)

[Занятие 10. Плоские и планарные графы СДАНО 36](#_Toc181199411)

[Задание 11 – коммивояжер 42](#_Toc181199412)

[Задание 14 Паросочетания 53](#_Toc181199413)

[Задание 16 ИТОГ 59](#_Toc181199414)

# Занятие 2 Проверено

Задание 1

Постройте граф, у которого радиус совпадает с диаметром (у графа должно быть не менее пяти вершин).



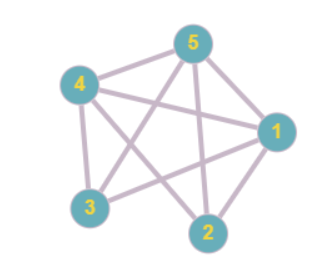
Задание 2

Постройте граф, у которого каждая вершина является и периферийной и центральной (у графа должно быть не менее пяти вершин).



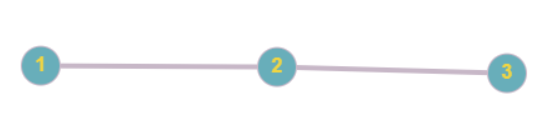
Задание 3

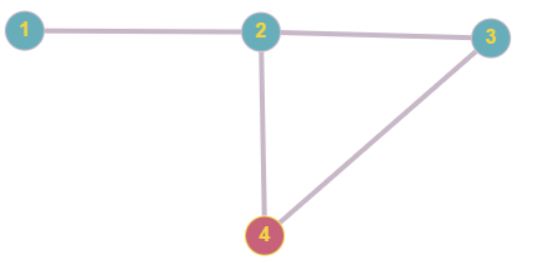
3. Постройте граф, у которого радиус равен единице, а диаметр двум.

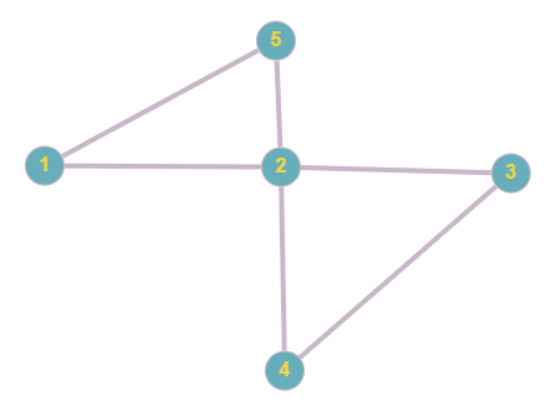


Задание 4

. Постройте графы с тремя, четырьмя и пятью вершинами, у которых центр состоит ровно из одной вершины (не использовать графы типа «звезда»).

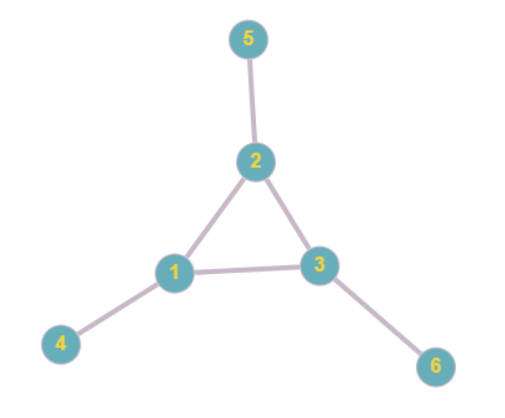






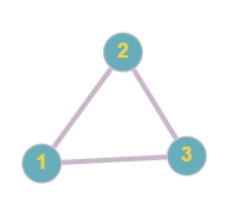
Задание 5

Постройте граф, у которого центр состоит ровно из трех вершин и не совпадает с множеством всех вершин.



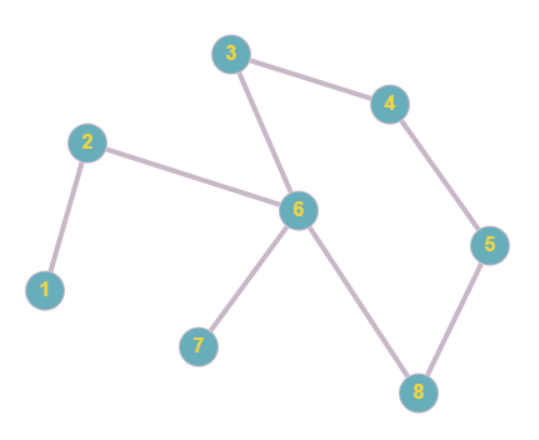
Задание 6

Постройте граф, такой, что центр состоит ровно из трех вершин и совпадает с множеством всех вершин



Задание 7

Построить простой граф на 8 вершинах и найти его радиус, диаметр, центральные и периферийные вершины (не использовать граф, являющийся простым циклом).



Радиус (2)

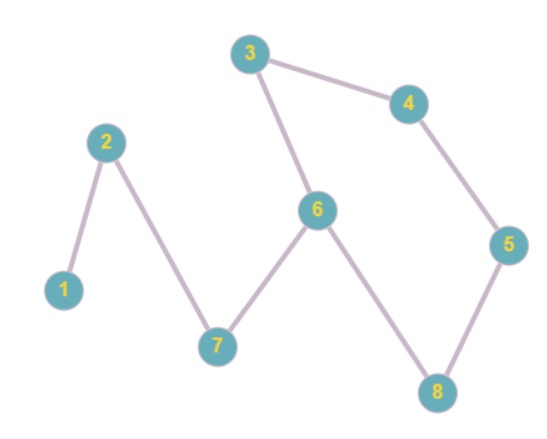
Диаметр 4

Центральные 6

Периферийные 1, 4, 5

Задание 8

Построить граф на 8 вершинах, такой что его радиус равен 3, а диаметр 5.



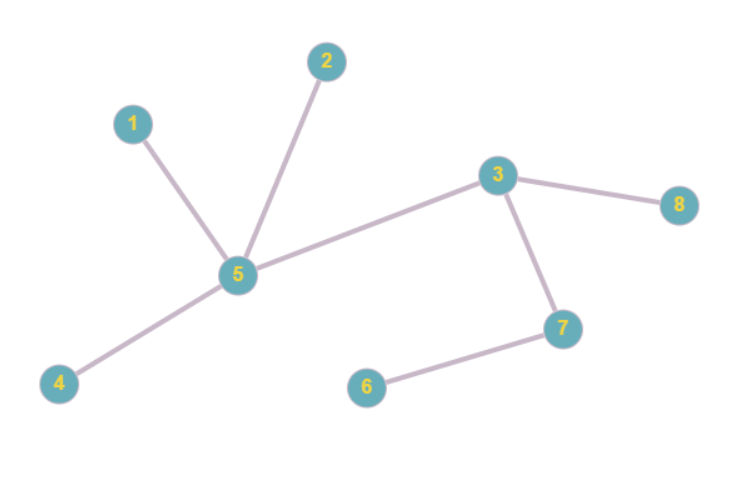
Задание 9

Какой наибольший диаметр может быть у графа с вершинами? Сколько имеется графов с таким диаметром (непомеченных)?

У графа с N вершинами диаметр максимально равен n-1, это линия из вершин, 1 возможный вариант

# ЗАНЯТИЕ 3 Проверено:

Задание 1



Задание 2:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 |
| 4 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| 5 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 0 |
| 7 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 8 | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 0 |

Задание 3:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| A | 1 | 0 | 0 | 0 | 1 | 0 | 0 | 0 |
| B | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 0 |
| C | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 |
| D | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 |
| E | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| F | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| G | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

Задание 4:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| 1 | 1 | 0 | 0 | 0 | -1 | 0 | 0 | 0 |
| 2 | 0 | 1 | 0 | 0 | -1 | 0 | 0 | 0 |
| 3 | 0 | 0 | 3 | 0 | -1 | 0 | -1 | -1 |
| 4 | 0 | 0 | 0 | 1 | -1 | 0 | 0 | 0 |
| 5 | -1 | -1 | -1 | -1 | 4 | 0 | 0 | 0 |
| 6 | 0 | 0 | 0 | 0 | 0 | 1 | -1 | 0 |
| 7 | 0 | 0 | -1 | 0 | 0 | -1 | 2 | 0 |
| 8 | 0 | 0 | -1 | 0 | 0 | 0 | 0 | 1 |

Задание 5

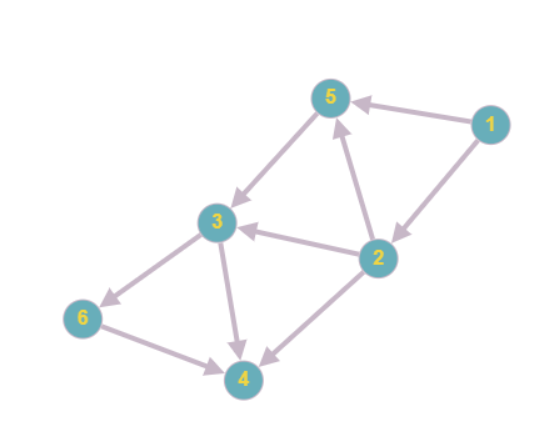
Без 1,4

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 2 | 3 | 5 | 6 | 7 | 8 |
| B | 1 | 0 | 1 | 0 | 0 | 0 |
| C | 0 | 1 | 1 | 0 | 0 | 0 |
| E | 0 | 1 | 0 | 0 | 0 | 1 |
| F | 0 | 1 | 0 | 0 | 1 | 0 |
| G | 0 | 0 | 0 | 1 | 1 | 0 |

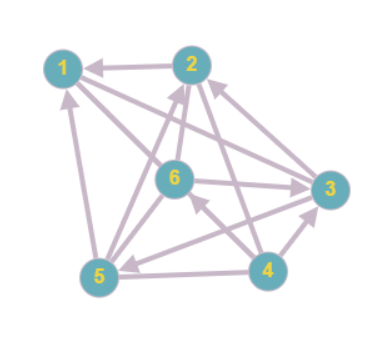
Без 1,5

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 |
| E | 0 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| F | 0 | 0 | 1 | 0 | 0 | 0 | 1 | 0 |
| G | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 |

Задание 6

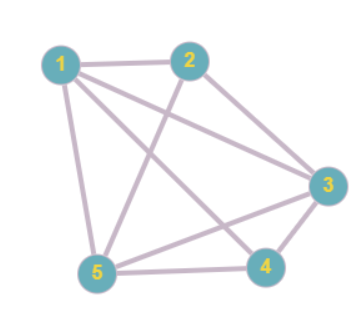


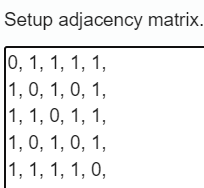
Задание 7

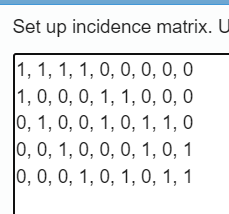


|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 | 6 |
| 1 | 0 | 0 | 1 | 1 | 0 | 1 |
| 2 | 1 | 0 | 0 | 1 | 0 | 1 |
| 3 | 1 | 1 | 0 | 0 | 1 | 0 |
| 4 | 1 | 1 | 1 | 0 | 1 | 1 |
| 5 | 1 | 1 | 0 | 1 | 0 | 1 |
| 6 | 1 | 1 | 1 | 0 | 1 | 0 |

Задание 8







|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | 1 | 2 | 3 | 4 | 5 |
| 1 | 4 | -1 | -1 | -1 | -1 |
| 2 | -1 | 3 | -1 | 0 | -1 |
| 3 | -1 | -1 | 4 | -1 | -1 |
| 4 | -1 | -1 | -1 | 4 | -1 |
| 5 | -1 | -1 | -1 | -1 | 4 |

Задание 9

A – полный и колесо

B – полный и колесо

C - двудольный

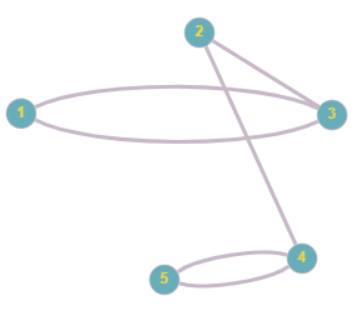
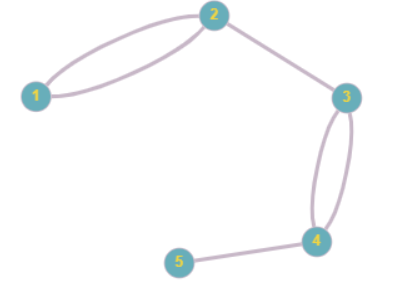
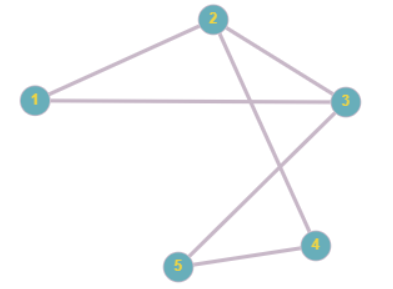
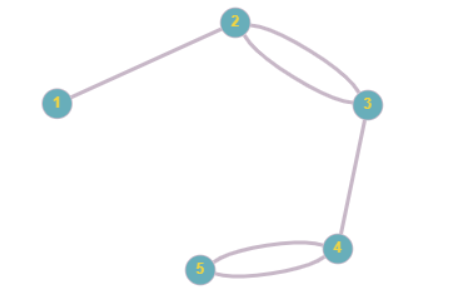
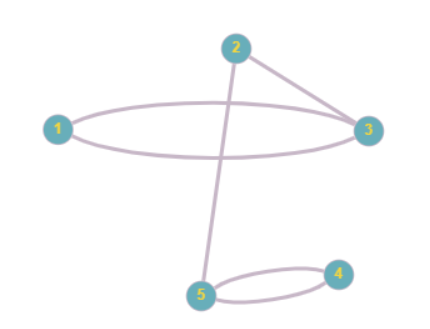
Изоморфны друг другу если существует взаимно однозначное соответствие между графами

Которое сохраняет отношение смежности

Отношение изоморфизма – отношение эквивалентности

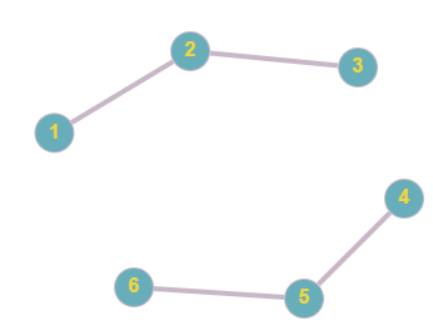
# Занятие 4: Изоморфизм СДАНО

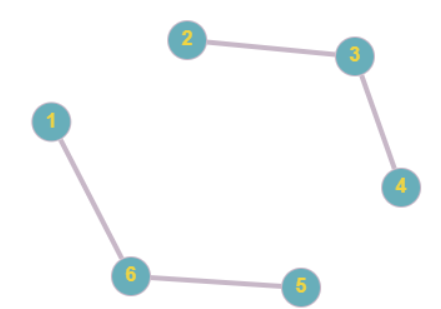
Задание 1 Определить, какой из графов (если такой есть) не изоморфен ни одному из остальных (в ответ указать букву списка для соответствующего графа). Для остальных графов определить, какому (каким) из графов они изоморфны.

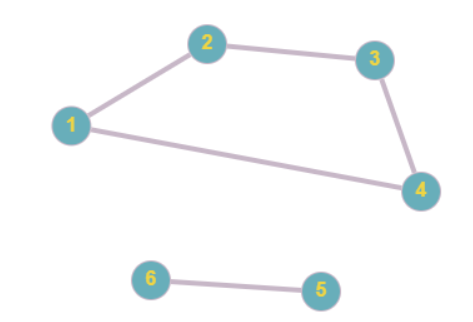
1. 
2. 
3. 
4. 
5. 

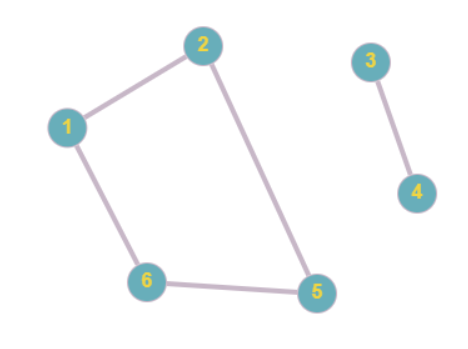
A и B и E изоморфны

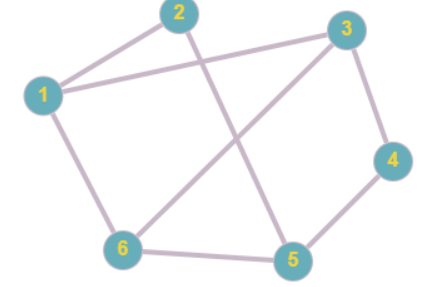
Задание 2 Самостоятельно подготовить задание, аналогичное задаче 1 (пять графов по 6 вершин). Для иллюстрации использовать одно из online средств визуализации графов. При оформлении решения этой задачи указать список графов, записанных в текстовой форме в виде перечисления ребер; скриншоты графов; ответ (буква списка выбранного графа (неизоморфного другим)); ответ по изоморфности остальных графов (например, “ граф a) изоморфен графу e) ”)

1.

2. 

3. 

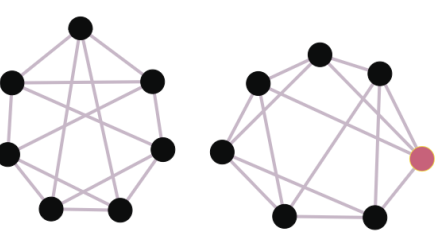
4. 

5. 

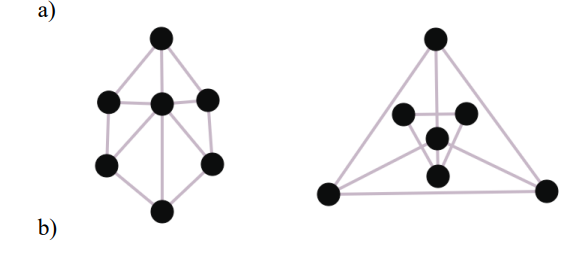
Задание 3 Докажите, что следующие графы G1 и G2 изоморфны, построив взаимно однозначное соответствие φ:V(G1)→V(G2)

1. 1 -> 1, 2->2,3->3,4->4,5->5,6->6,7->7
2. x1 -> 1, x2->3,x3->2,y1->5,y2->6,y3->4
3. x1 -> A, x2->E,x3->C,y1->B,y2->D,y3->F

Задание 4 4. Докажите, что следующие графы не являются изоморфными:



Они не изоморфны с силу метода построения и нечетности кол-во вершин (левый 4 вершины с отсутствием 2. СЧИТАЙ ЦИКЛЫ ДЛИНЫ 3

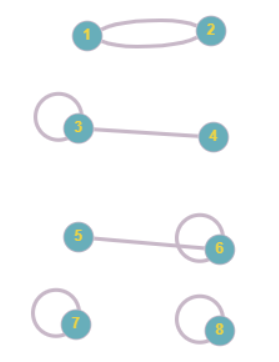


Тут 2 компоненты связности в правом, в левом 1

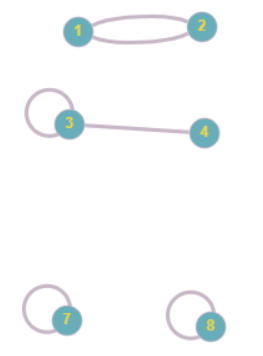


Если убрать нижнюю вершину на фотке справа станет 2 графа, слева такой веришны нету.

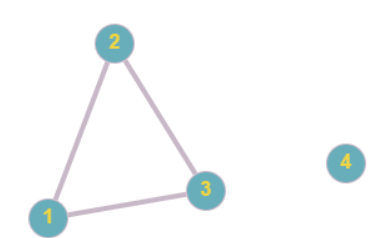
Задание 5 Нарисуйте все возможные графы с двумя вершинами и двумя ребрами (петли разрешены). Нарисуйте все неизоморфные графы с двумя вершинами и двумя ребрами.

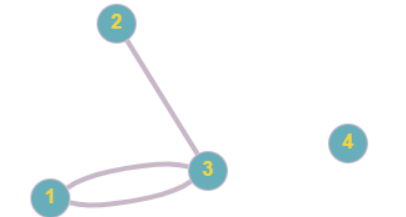
Все 

Неизоморфные

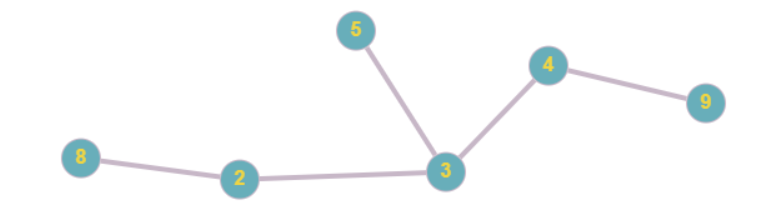
****

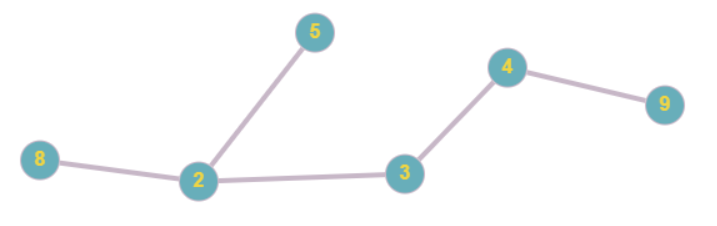
Задание 6: Приведите пример двух неизоморфных графов с одинаковым числом вершин, ребер и компонент связности.





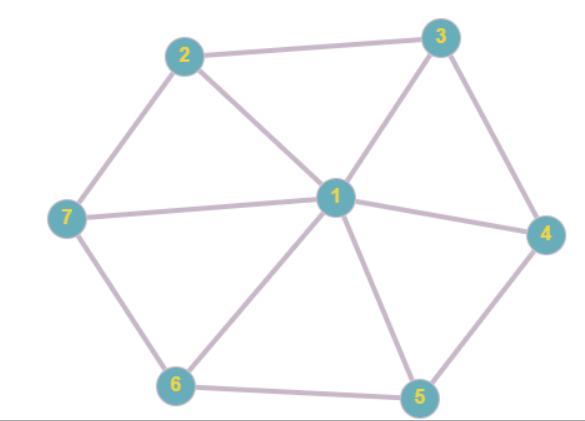
Задание 7: Приведите пример двух неизоморфных графов с одинаковым числом вершин, ребер, компонент связности и одинаковым набором степеней вершин.

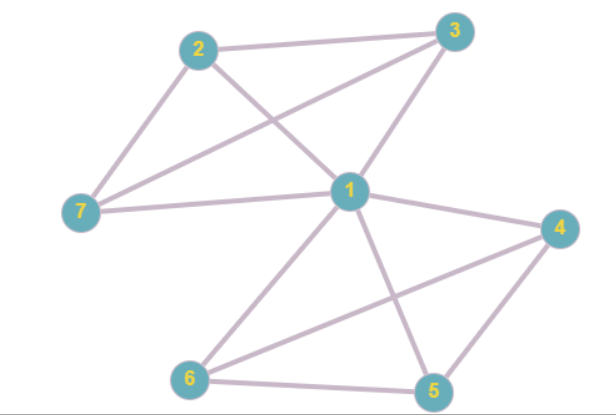




Задание 8: это верно в силу наличия общего множества (универсального, полного графа) и свойств биекции

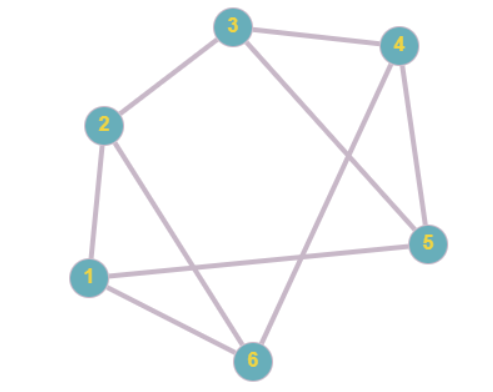
Задание 9

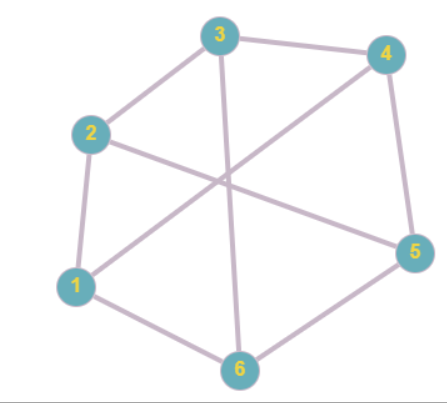




Задание 10 Приведите пример двух неизоморфных кубических графов

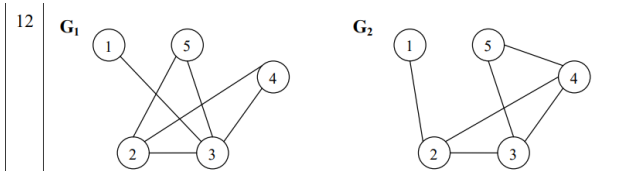
на 6 вершинах с 9 ребрами.



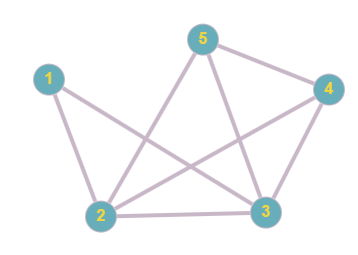


# Занятие 5 Операции над графами СДАНО

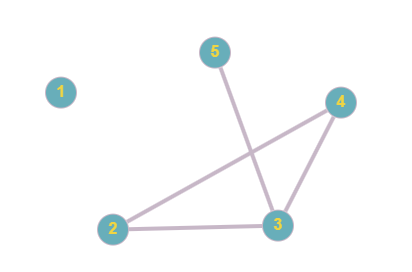
Задание 1: Выбрать 2 графа из предложенных ниже вариантов



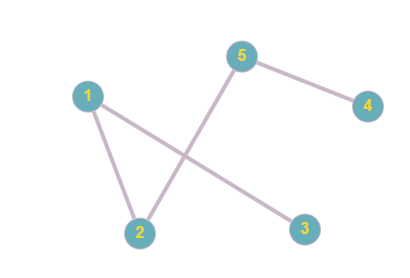
Задание 2: Построить граф, равный объединению исходных графов.



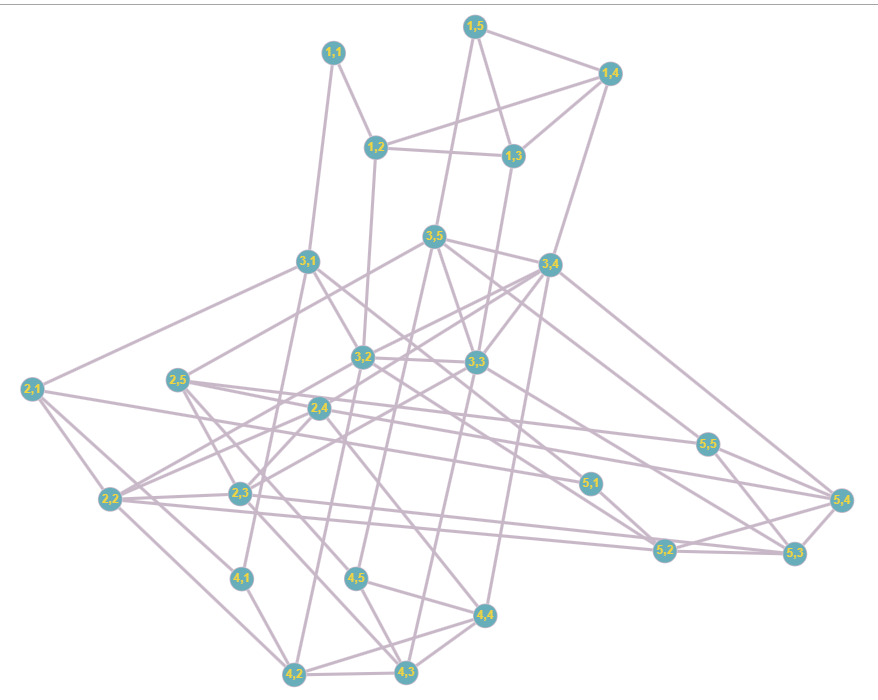
Задание 3: Построить граф, равный пересечению исходных графов.



Задание 4: Построить кольцевую сумму исходных графов.

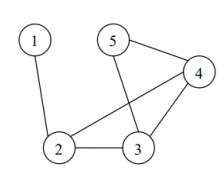


Задание 5: Построить декартово произведение графов

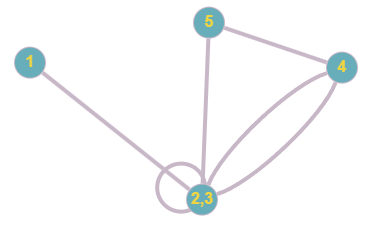


Задание 6: Для одного из исходных графов выполните операцию отождествления двух произвольных вершин

Было

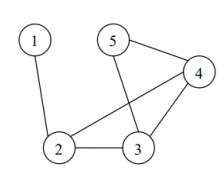


отождествлю 2 – 3

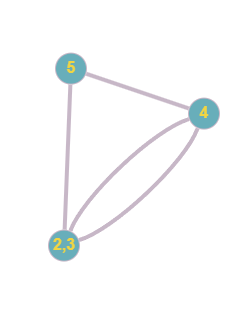


Задание 7: Для одного из исходных графов выполните операцию стягивания двух произвольных ребер

Было

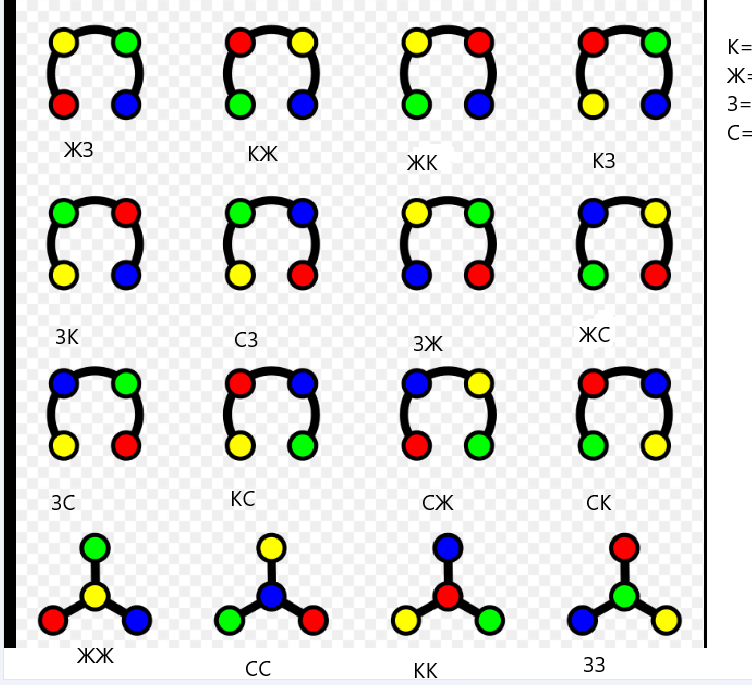


Стягивание ребра (2,3) и (1,2)

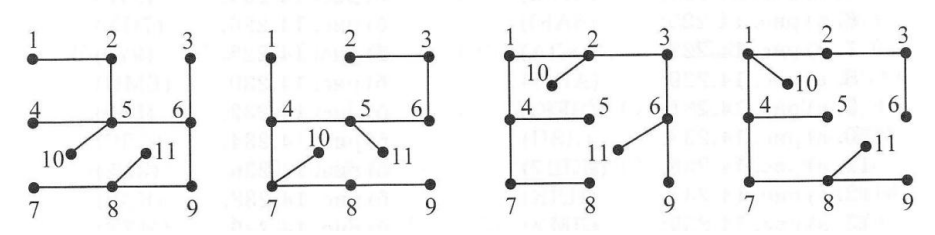


# Занятие 6. Код Прюфера СДАНО

Задание 1: Выпишите коды Прюфера всех помеченных деревьев с четырьмя вершинами и убедитесь, что каждая последовательность длины два из номеров вершин 1, 2, 3, 4 встречается среди этих кодов ровно один раз.



Задание 2: Найдите код дерева методом Прюфера



1 – 2 5 6 5 8 5 6 9 8

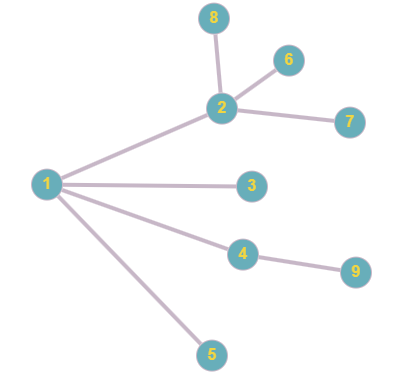
2 – 4 3 6 5 8 7 7 4 5

3 – 4 7 4 1 2 6 2 3 6

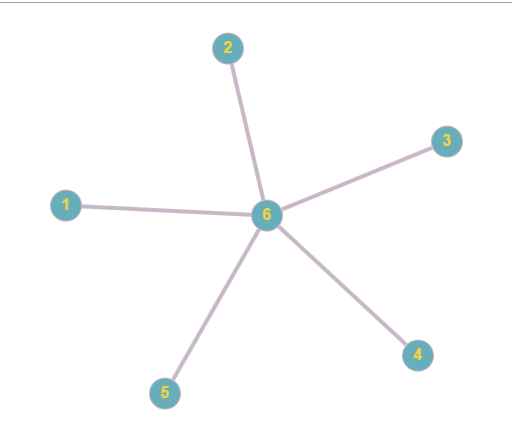
4 – 4 3 2 1 8 1 4 7 8

Задание 3:

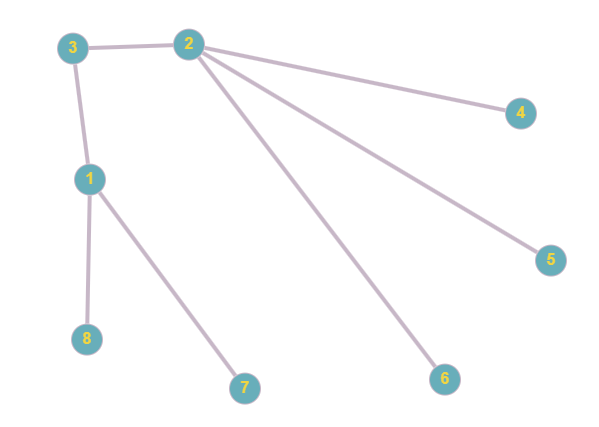
1)



2)



3)



Задание 4: Дерево задано следующим кодом Прюфера: 667767677

а) сколько ребер соединяют вершины 2 и 8, 4 и 9

dist(2,8) = 3, dist(4,9) = 3

б) укажите степени вершин с номерами 5, 6,11

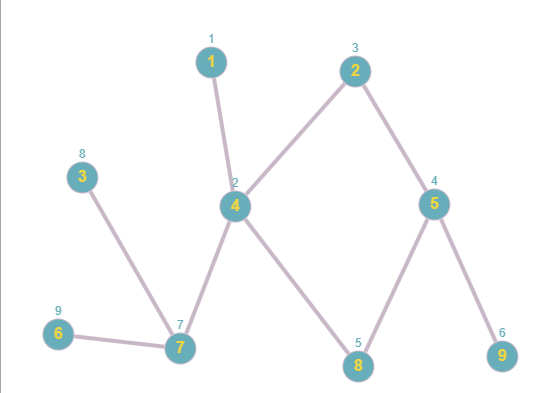
deg(5)=1,deg(11)=1,deg(6)=5

Задание 5: Написать программу, которая по заданному коду Прюфера (в качестве тестовых примеров взять примеры из задания 1) восстанавливает дерево, то есть находит списки смежности его вершин.

Lesson 6 – ex\_5()

# Занятие 7. Алгоритм поиска в глубину и поиска в ширину СДАНО

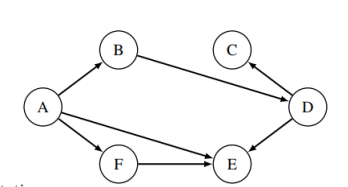
Задача 1.



В Глубину: 1-4-2-5-8-9-7-3-6

В ширину: 1-4-2-7-8-5-3-6-9

Задача 2:



В Глубину: A-B-D-C-E-F

В ширину: A-B-E-F-D-E-C

Задача 3: Полного

Задача 4: Простой цикл из 3-х вершин

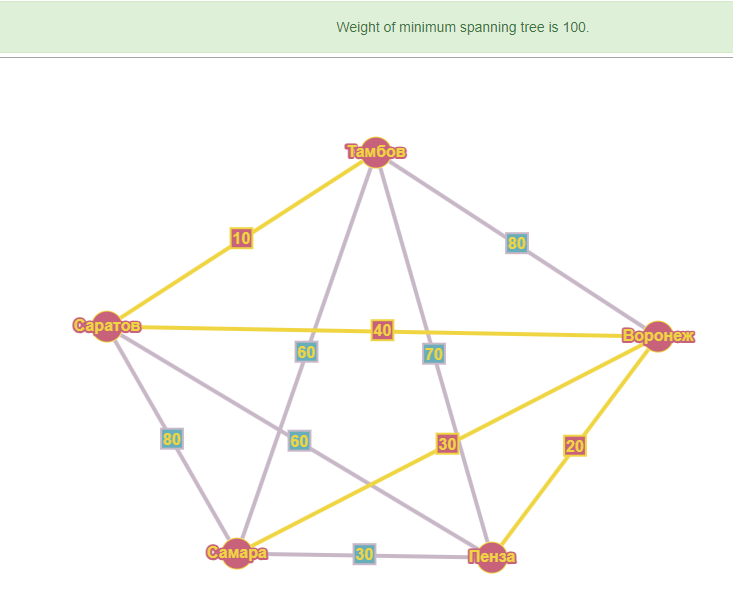
Задача 5: Полного

Задание 6: Реализуйте программу, в которой выполняется алгоритм обхода графа на основе поиска в глубину. Lesson 7 -> ex\_6()

Задание 7 Реализуйте программу, в которой выполняется алгоритм обхода графа на основе поиска в ширину. Lesson 7 -> ex\_7()

Задание 8 Используйте обход графа в ширину для определения всех вершин графа, находящихся на фиксированном расстоянии d от данной вершины.. Lesson 7 -> ex\_8()

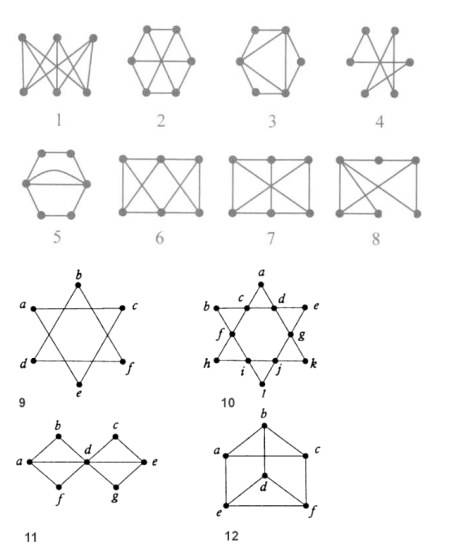
# Занятие 8 Алгоритм поиска минимального остовного дерева: СДАНО

Задание 1: 

Задание 2: Даны точки на плоскости, являющиеся вершинами полного графа. Вес ребра равен расстоянию между точками, соответствующими концам этого ребра. Требуется в этом графе найти остовное дерево минимального веса. Lesson10 -> ex\_2()

# Занятие 9. Эйлеровы и гамильтоновы графы

Задание 1 Укажите сначала номера всех эйлеровых графов в порядке возрастания, а затем – номера всех полуэйлеровых (также в порядке возрастания)



Эйлеров цикл: 3, 4, 5, 10

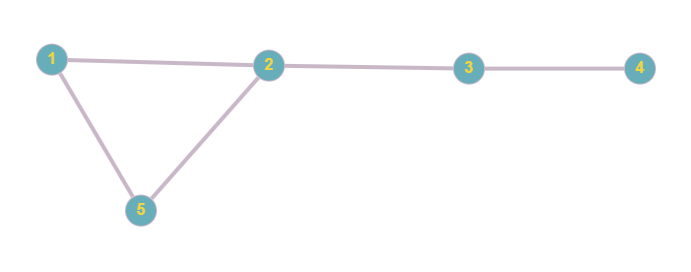
Эйлеров путь: 8,11

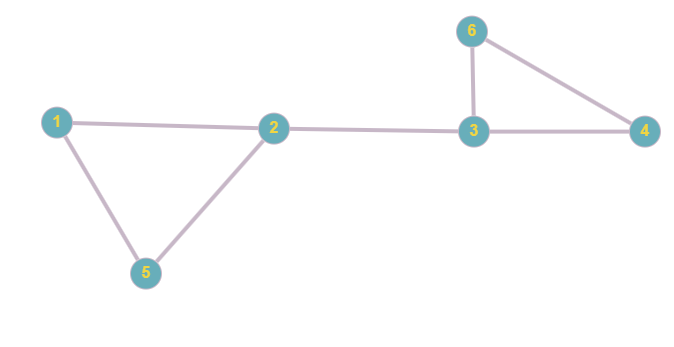
Ничего из: 9, 1, 2, 7, 12, 6

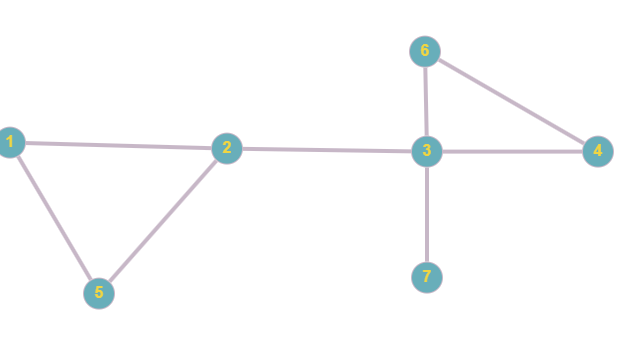
Задание 2

Полуэйлеровы

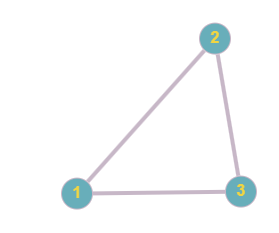


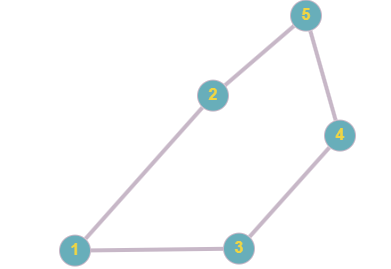


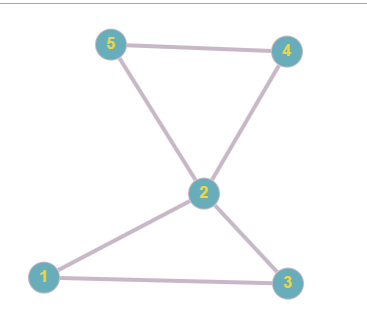


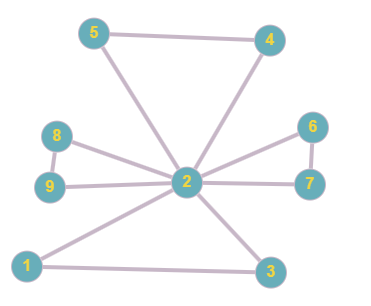


Эйлеров







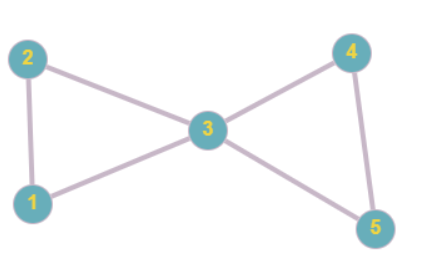


Задание 3 – Имеется кусок проволоки длиной 12 сантиметров. На какое минимальное количество кусков его следует разрезать, чтобы из этих кусков можно было бы изготовить каркас кубика размерами 1 × 1 × 1 при условии, что проволоку в процессе изготовления кубиков можно сгибать?

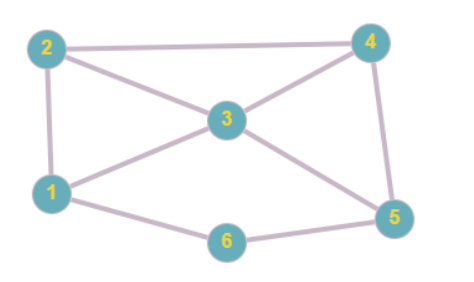
1 разрез, нужно 2 куска проволки.

Задание 4 Приведите пример эйлерова графа, не являющегося гамильтоновым и гамильтонова графа, не являющегося эйлеровым. Граф имеет 6 вершин.

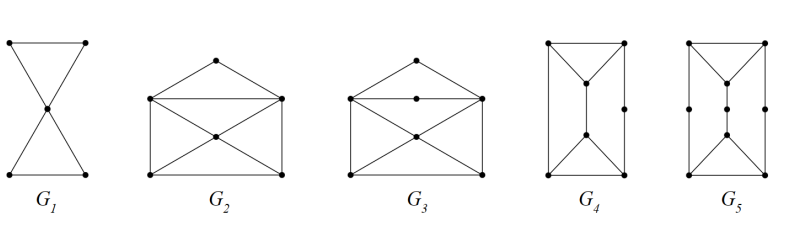
Эйлеров но не гамильтонов



Гамильтонов но не эйлеров

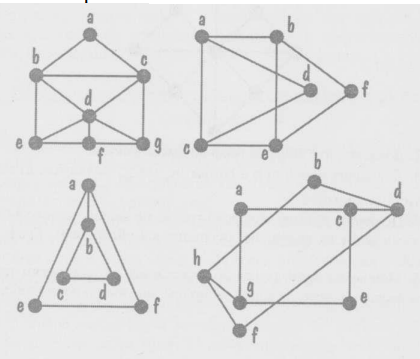


Задание 5 Среди графов Gi, показанных на рисунке, указать графы, в которых гамильтоновы циклы существуют. Постарайтесь для себя привести доказательство гамильтоновости или негамильтоновости соответствующих графов.



Есть гам цикл – 2, 4

Задание 6 Найти гамильтонов цикл (цепь) в следующих графах, если он существует, с помощью переборного алгоритма:



Есть путь в левом верхнем, правом верхнем.

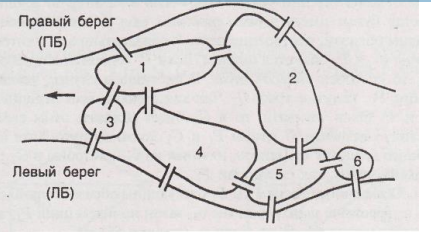
Задание 7 При каких m и n следующие графы являются а) эйлеровыми б) гамильтоновыми: Kn.Kmn,Wn.

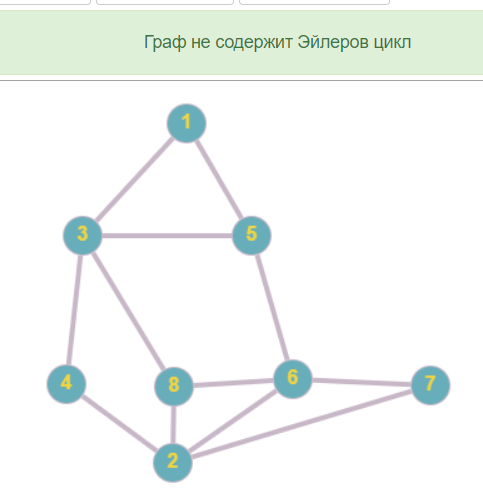
А) Kn всегда гамильтонов, эйлеров для нечетных n

Б) Kmn – гамильтонов когда m=n, когда m,n четные то эйлеров

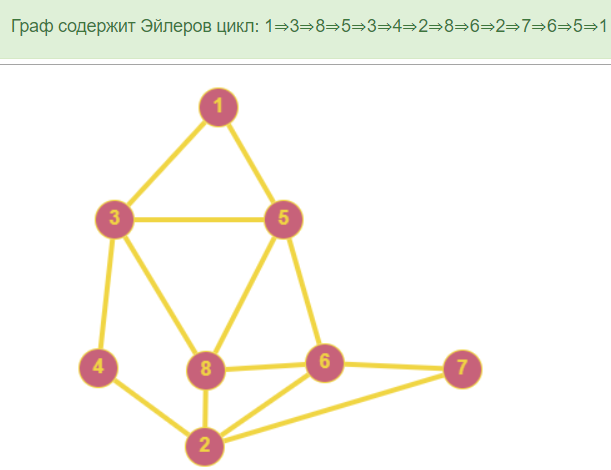
В) Wn – всегда гамильтонов, никогда эйлеров

Задание 8 Решите следующую задачу: шесть островов на реке соединены мостами, как показано на рисунке. Можно ли, начав прогулку на одном из островов, пройти по каждому из мостиков ровно один раз и вернуться на тот же остров? В случае отрицательного ответа определите, сколько мостиков и между каким островами нужно построить, чтобы такая прогулка стала возможной.





Нужно добавить 1 ребро 8-5

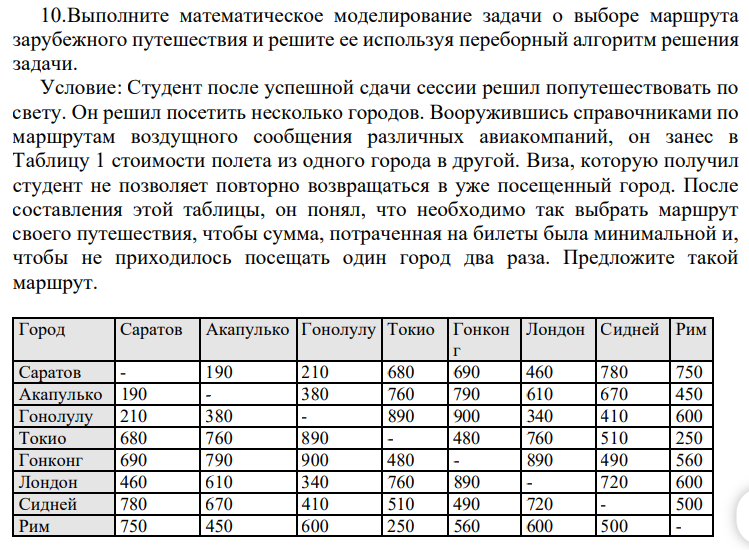


Задание 9

Формат входных данных: В первой строке указаны два числа разделенных пробелом: число вершин и число ребер. В следующих строках указаны пары вершин, соединенных ребром. Выполняются ограничения: 2≤число вершин≤1000,0≤число ребер≤1000. Формат выходных данных: Одно слово: NONE, если в графе нет эйлерова цикла, или список вершин в порядке обхода эйлерова цикла, если он есть

Lesson9 -> ex\_9()

Задание 10

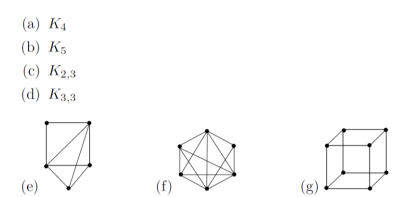


Lesson 9 -> ex\_10()



# Занятие 10. Плоские и планарные графы СДАНО

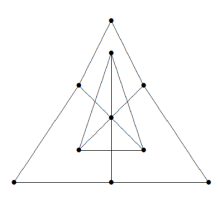
Задание 1: Определите, какие из приведенных ниже графов являются планарными, а какие не планарными. Объясните почему.

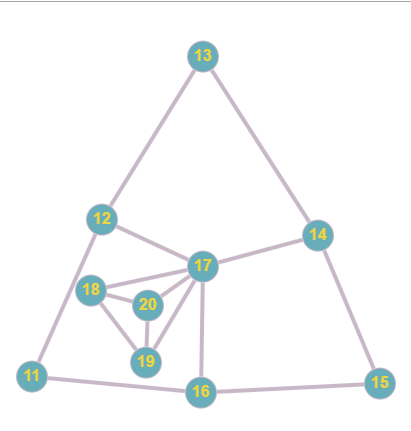


Планарные a, c, e, g

Не планарные b, c, f (св-во эйлера 13<6\*3-6)

Задание 2: Покажите, что следующий граф планарен, построив его плоское изображение:





Задание 3:

Пусть G связный граф с числом вершин больше или равным 11. Докажите, что либо граф, либо его дополнение является непланарным.

Для доказательства этого утверждения мы можем использовать теорему Куратовского, которая утверждает следующее: граф планарен тогда и только тогда, когда он не содержит подграф, гомеоморфный K5 (полный граф на 5 вершинах) или подграф, гомеоморфный K3,3 (поделенный граф на 2 непересекающихся троек вершин).  
  
Итак, рассмотрим G как связный граф с числом вершин больше или равным 11. Теперь предположим обратное: исходные G и его дополнение оба планарны. Поскольку G и его дополнение являются планарными, они не содержат ни K5-содержащих подграфов ни K3.3-содержащих подграфов.  
  
Теперь рассмотрим количество ребер в G и его дополнении. Сумма количества ребер в G и его дополнении равна количеству ребер полного графа Kn (n - количество вершин). Так как n более или равно 11 (как предполагается), то n(n-1)/2 (число всех возможных ребер) будет больше чем сумма количеств ребр в двух планарных гризе.  
  
Следовательно хотя бы один из двух гризев муст быть не планаранным , что завёршает наше предположение .

Задание 4:

A - 2

B – нет

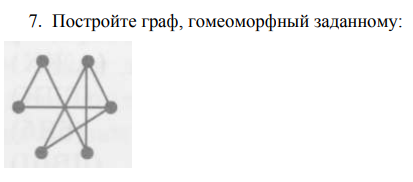
C – нет

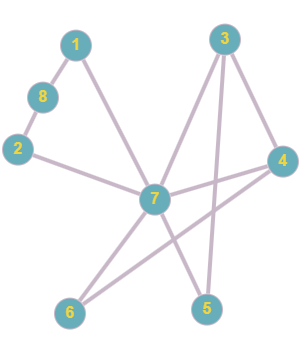
D – 5, 5

Задание 5: является

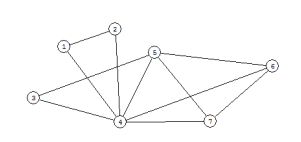
Задание 6: по формуле эйлера 37 10-45+x=2

Задание 7:

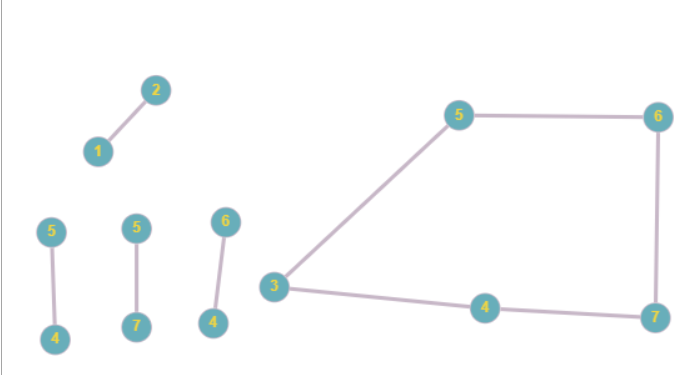


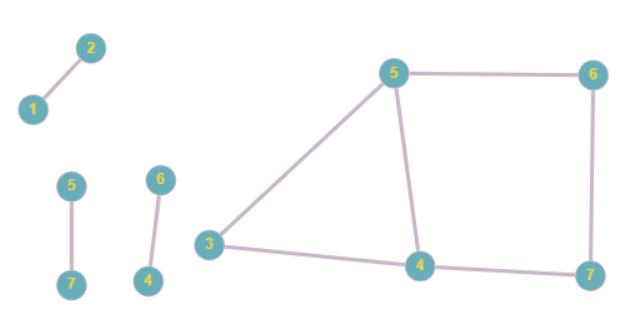


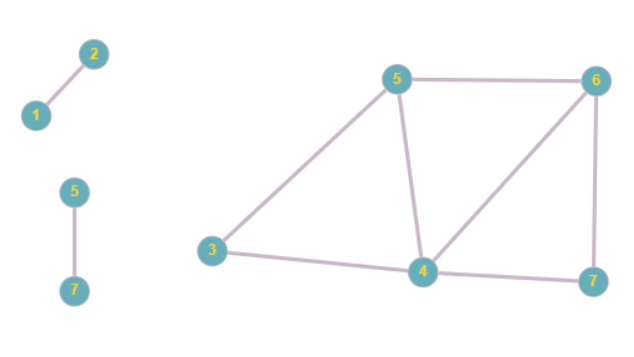
Задание 8: Если возможно, постройте плоскую укладку следующих графов, используя гамма-алгоритм. Подробно опишите каждый шаг алгоритма.

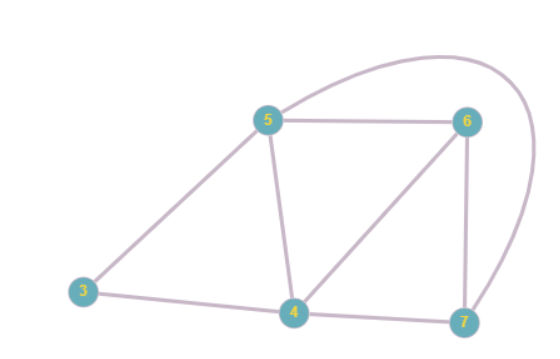


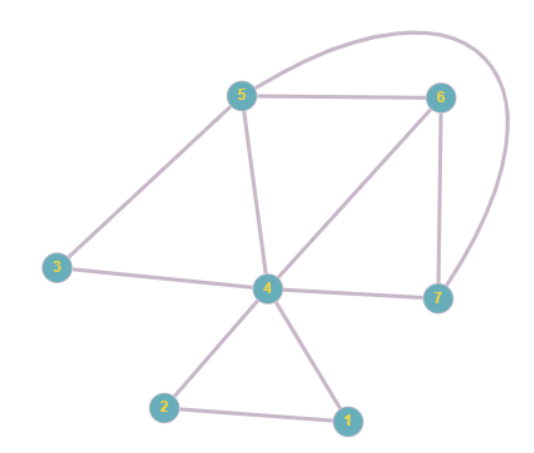
Гамма алгоритм:



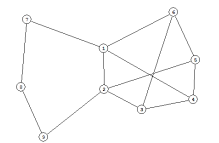




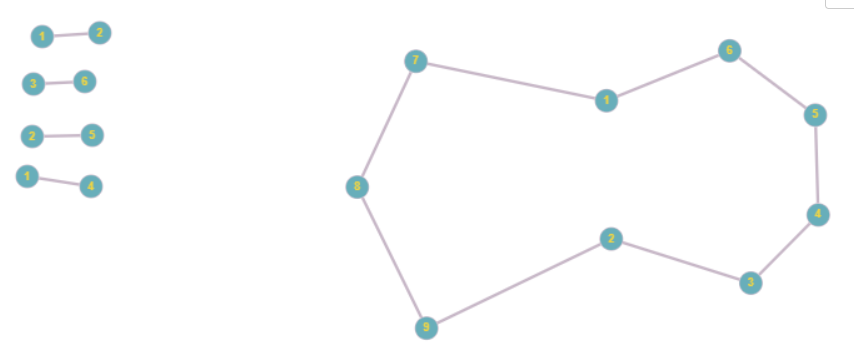


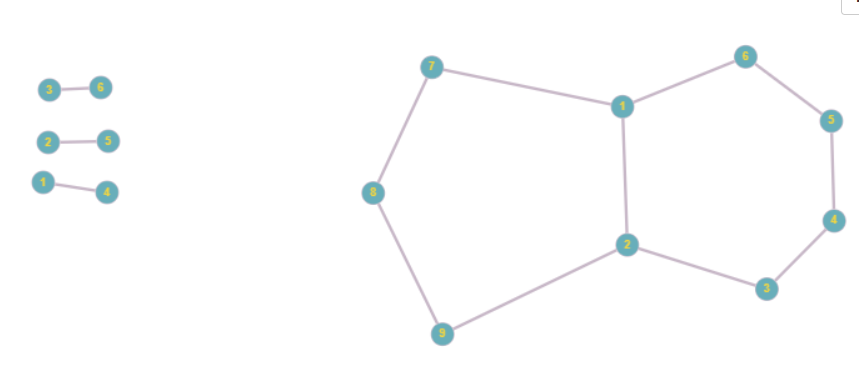


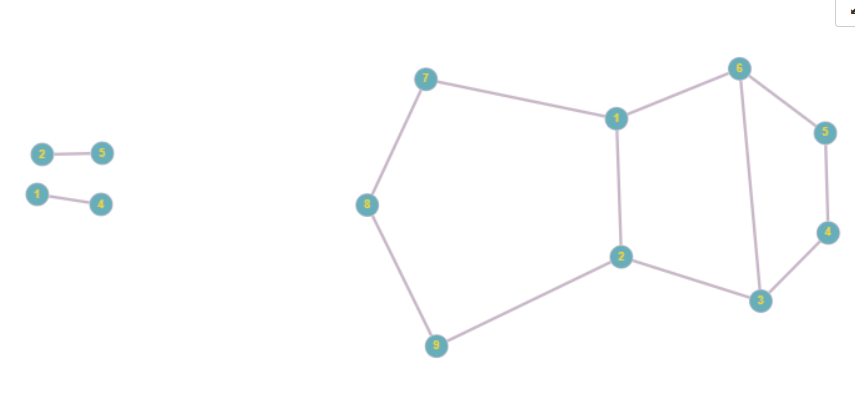
2. Это не возможно тк K3,3



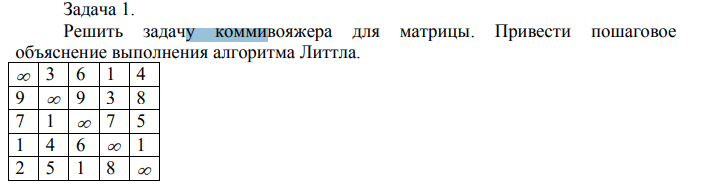
Гамма-алгоритм







# Задание 11 – коммивояжер



Возьмем в качестве произвольного маршрута:  
X0 = (1,2);(2,3);(3,4);(4,5);(5,1)  
Тогда F(X0) = 3 + 9 + 7 + 1 + 2 = 22  
Для определения нижней границы множества воспользуемся **операцией редукции** или приведения матрицы по строкам, для чего необходимо в каждой строке матрицы D найти минимальный элемент.  
di = min(j) dij

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | di |
| **1** | M | 3 | 6 | 1 | 4 | 1 |
| **2** | 9 | M | 9 | 3 | 8 | 3 |
| **3** | 7 | 1 | M | 7 | 5 | 1 |
| **4** | 1 | 4 | 6 | M | 1 | 1 |
| **5** | 2 | 5 | 1 | 8 | M | 1 |

Затем вычитаем di из элементов рассматриваемой строки. В связи с этим во вновь полученной матрице в каждой строке будет как минимум один ноль.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** |
| **1** | M | 2 | 5 | 0 | 3 |
| **2** | 6 | M | 6 | 0 | 5 |
| **3** | 6 | 0 | M | 6 | 4 |
| **4** | 0 | 3 | 5 | M | 0 |
| **5** | 1 | 4 | 0 | 7 | M |

Такую же операцию редукции проводим по столбцам, для чего в каждом столбце находим минимальный элемент:  
dj = min(i) dij

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** |
| **1** | M | 2 | 5 | 0 | 3 |
| **2** | 6 | M | 6 | 0 | 5 |
| **3** | 6 | 0 | M | 6 | 4 |
| **4** | 0 | 3 | 5 | M | 0 |
| **5** | 1 | 4 | 0 | 7 | M |
| dj | 0 | 0 | 0 | 0 | 0 |

После вычитания минимальных элементов получаем полностью редуцированную матрицу, где величины di и dj называются **константами приведения**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** |
| **1** | M | 2 | 5 | 0 | 3 |
| **2** | 6 | M | 6 | 0 | 5 |
| **3** | 6 | 0 | M | 6 | 4 |
| **4** | 0 | 3 | 5 | M | 0 |
| **5** | 1 | 4 | 0 | 7 | M |

Сумма констант приведения определяет нижнюю границу H:  
H = ∑di + ∑dj  
H = 1+3+1+1+1+0+0+0+0+0 = 7  
Элементы матрицы dij соответствуют расстоянию от пункта i до пункта j.  
Поскольку в матрице n городов, то D является матрицей nxn с неотрицательными элементами dij ≥ 0  
Каждый допустимый маршрут представляет собой цикл, по которому коммивояжер посещает город только один раз и возвращается в исходный город.  
Длина маршрута определяется выражением:  
F(Mk) = ∑dij  
Причем каждая строка и столбец входят в маршрут только один раз с элементом dij.  
**Шаг №1**.  
**Определяем ребро ветвления** и разобьем все множество маршрутов относительно этого ребра на два подмножества (i,j) и (i\*,j\*).  
С этой целью для всех клеток матрицы с нулевыми элементами заменяем поочередно нули на М(бесконечность) и определяем для них сумму образовавшихся констант приведения, они приведены в скобках.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | di |
| **1** | M | 2 | 5 | 0(2) | 3 | 2 |
| **2** | 6 | M | 6 | 0(5) | 5 | 5 |
| **3** | 6 | **0(6)** | M | 6 | 4 | 4 |
| **4** | 0(1) | 3 | 5 | M | 0(3) | 0 |
| **5** | 1 | 4 | 0(6) | 7 | M | 1 |
| dj | 1 | 2 | 5 | 0 | 3 | 0 |

d(1,4) = 2 + 0 = 2; d(2,4) = 5 + 0 = 5; d(3,2) = 4 + 2 = 6; d(4,1) = 0 + 1 = 1; d(4,5) = 0 + 3 = 3; d(5,3) = 1 + 5 = 6;  
Наибольшая сумма констант приведения равна (4 + 2) = 6 для ребра (3,2), следовательно, множество разбивается на два подмножества (3,2) и (3\*,2\*).  
**Исключение ребра** (3,2) проводим путем замены элемента d32 = 0 на M, после чего осуществляем очередное приведение матрицы расстояний для образовавшегося подмножества (3\*,2\*), в результате получим редуцированную матрицу.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | di |
| **1** | M | 2 | 5 | 0 | 3 | 0 |
| **2** | 6 | M | 6 | 0 | 5 | 0 |
| **3** | 6 | M | M | 6 | 4 | 4 |
| **4** | 0 | 3 | 5 | M | 0 | 0 |
| **5** | 1 | 4 | 0 | 7 | M | 0 |
| dj | 0 | 2 | 0 | 0 | 0 | 6 |

Нижняя граница гамильтоновых циклов этого подмножества:  
H(3\*,2\*) = 7 + 6 = 13  
**Включение ребра** (3,2) проводится путем исключения всех элементов 3-ой строки и 2-го столбца, в которой элемент d23 заменяем на М, для исключения образования негамильтонова цикла.  
В результате получим другую сокращенную матрицу (4 x 4), которая подлежит операции приведения.  
После операции приведения сокращенная матрица будет иметь вид:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **3** | **4** | **5** | di |
| **1** | M | 5 | 0 | 3 | 0 |
| **2** | 6 | M | 0 | 5 | 0 |
| **4** | 0 | 5 | M | 0 | 0 |
| **5** | 1 | 0 | 7 | M | 0 |
| dj | 0 | 0 | 0 | 0 | 0 |

Сумма констант приведения сокращенной матрицы:  
∑di + ∑dj = 0  
Нижняя граница подмножества (3,2) равна:  
H(3,2) = 7 + 0 = 7 ≤ 13  
Поскольку нижняя граница этого подмножества (3,2) меньше, чем подмножества (3\*,2\*), то ребро (3,2) включаем в маршрут с новой границей H = 7.  
**Шаг №2**.  
**Определяем ребро ветвления**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **3** | **4** | **5** | di |
| **1** | M | 5 | 0(3) | 3 | 3 |
| **2** | 6 | M | 0(5) | 5 | 5 |
| **4** | 0(1) | 5 | M | 0(3) | 0 |
| **5** | 1 | **0(6)** | 7 | M | 1 |
| dj | 1 | 5 | 0 | 3 | 0 |

d(1,4) = 3 + 0 = 3; d(2,4) = 5 + 0 = 5; d(4,1) = 0 + 1 = 1; d(4,5) = 0 + 3 = 3; d(5,3) = 1 + 5 = 6;  
max: d(5,3)=6.  
**Исключение ребра** (5,3): d53=M.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **3** | **4** | **5** | di |
| **1** | M | 5 | 0 | 3 | 0 |
| **2** | 6 | M | 0 | 5 | 0 |
| **4** | 0 | 5 | M | 0 | 0 |
| **5** | 1 | M | 7 | M | 1 |
| dj | 0 | 5 | 0 | 0 | 6 |

H(5\*,3\*) = 7 + 6 = 13  
**Включение ребра** (5,3): d35=М.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **1** | **4** | **5** | di |
| **1** | M | 0 | 3 | 0 |
| **2** | 6 | 0 | 5 | 0 |
| **4** | 0 | M | 0 | 0 |
| dj | 0 | 0 | 0 | 0 |

∑di + ∑dj = 0  
H(5,3) = 7 + 0 = 7 ≤ 13  
Чтобы исключить подциклы, запретим следующие переходы: (2,5),  
Ребро (5,3) включаем в маршрут с новой границей H=7.  
**Шаг №3**.  
**Определяем ребро ветвления**.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **1** | **4** | **5** | di |
| **1** | M | 0(3) | 3 | 3 |
| **2** | 6 | 0(6) | M | 6 |
| **4** | **0(6)** | M | 0(3) | 0 |
| dj | 6 | 0 | 3 | 0 |

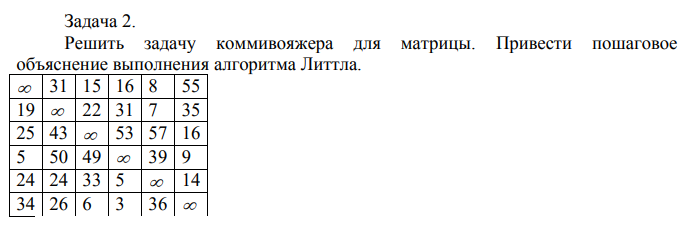
d(1,4) = 3 + 0 = 3; d(2,4) = 6 + 0 = 6; d(4,1) = 0 + 6 = 6; d(4,5) = 0 + 3 = 3;  
max: d(4,1)=6.  
**Исключение ребра** (4,1): d41=M.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **1** | **4** | **5** | di |
| **1** | M | 0 | 3 | 0 |
| **2** | 6 | 0 | M | 0 |
| **4** | M | M | 0 | 0 |
| dj | 6 | 0 | 0 | 6 |

H(4\*,1\*) = 7 + 6 = 13  
**Включение ребра** (4,1): d14=М.

|  |  |  |  |
| --- | --- | --- | --- |
| **i j** | **4** | **5** | di |
| **1** | M | 3 | 3 |
| **2** | 0 | M | 0 |
| dj | 0 | 3 | 6 |

∑di + ∑dj = 6  
H(4,1) = 7 + 6 = 13 ≤ 13  
Ребро (4,1) включаем в маршрут с новой границей H=13.  
В соответствии с этой матрицей включаем в гамильтонов маршрут ребра (1,5) и (2,4).  
В результате по дереву ветвлений гамильтонов цикл образуют ребра:  
(3,2), (2,4), (4,1), (1,5), (5,3),  
Длина маршрута равна F(Mk) = 10



Возьмем в качестве произвольного маршрута:  
X0 = (1,2);(2,3);(3,4);(4,5);(5,6);(6,1)  
Тогда F(X0) = 31 + 22 + 53 + 39 + 14 + 34 = 193  
Для определения нижней границы множества воспользуемся **операцией редукции** или приведения матрицы по строкам, для чего необходимо в каждой строке матрицы D найти минимальный элемент.  
di = min(j) dij

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | M | 31 | 15 | 16 | 8 | 55 | 8 |
| **2** | 19 | M | 22 | 31 | 7 | 35 | 7 |
| **3** | 25 | 43 | M | 53 | 57 | 16 | 16 |
| **4** | 5 | 50 | 49 | M | 39 | 9 | 5 |
| **5** | 24 | 24 | 33 | 5 | M | 14 | 5 |
| **6** | 34 | 26 | 6 | 3 | 36 | M | 3 |

Затем вычитаем di из элементов рассматриваемой строки. В связи с этим во вновь полученной матрице в каждой строке будет как минимум один ноль.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | M | 23 | 7 | 8 | 0 | 47 |
| **2** | 12 | M | 15 | 24 | 0 | 28 |
| **3** | 9 | 27 | M | 37 | 41 | 0 |
| **4** | 0 | 45 | 44 | M | 34 | 4 |
| **5** | 19 | 19 | 28 | 0 | M | 9 |
| **6** | 31 | 23 | 3 | 0 | 33 | M |

Такую же операцию редукции проводим по столбцам, для чего в каждом столбце находим минимальный элемент:  
dj = min(i) dij

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | M | 23 | 7 | 8 | 0 | 47 |
| **2** | 12 | M | 15 | 24 | 0 | 28 |
| **3** | 9 | 27 | M | 37 | 41 | 0 |
| **4** | 0 | 45 | 44 | M | 34 | 4 |
| **5** | 19 | 19 | 28 | 0 | M | 9 |
| **6** | 31 | 23 | 3 | 0 | 33 | M |
| dj | 0 | 19 | 3 | 0 | 0 | 0 |

После вычитания минимальных элементов получаем полностью редуцированную матрицу, где величины di и dj называются **константами приведения**.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** |
| **1** | M | 4 | 4 | 8 | 0 | 47 |
| **2** | 12 | M | 12 | 24 | 0 | 28 |
| **3** | 9 | 8 | M | 37 | 41 | 0 |
| **4** | 0 | 26 | 41 | M | 34 | 4 |
| **5** | 19 | 0 | 25 | 0 | M | 9 |
| **6** | 31 | 4 | 0 | 0 | 33 | M |

Сумма констант приведения определяет нижнюю границу H:  
H = ∑di + ∑dj  
H = 8+7+16+5+5+3+0+19+3+0+0+0 = 66  
Элементы матрицы dij соответствуют расстоянию от пункта i до пункта j.  
Поскольку в матрице n городов, то D является матрицей nxn с неотрицательными элементами dij ≥ 0  
Каждый допустимый маршрут представляет собой цикл, по которому коммивояжер посещает город только один раз и возвращается в исходный город.  
Длина маршрута определяется выражением:  
F(Mk) = ∑dij  
Причем каждая строка и столбец входят в маршрут только один раз с элементом dij.  
**Шаг №1**.  
**Определяем ребро ветвления** и разобьем все множество маршрутов относительно этого ребра на два подмножества (i,j) и (i\*,j\*).  
С этой целью для всех клеток матрицы с нулевыми элементами заменяем поочередно нули на М(бесконечность) и определяем для них сумму образовавшихся констант приведения, они приведены в скобках.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | M | 4 | 4 | 8 | 0(4) | 47 | 4 |
| **2** | 12 | M | 12 | 24 | 0(12) | 28 | 12 |
| **3** | 9 | 8 | M | 37 | 41 | 0(12) | 8 |
| **4** | **0(13)** | 26 | 41 | M | 34 | 4 | 4 |
| **5** | 19 | 0(4) | 25 | 0(0) | M | 9 | 0 |
| **6** | 31 | 4 | 0(4) | 0(0) | 33 | M | 0 |
| dj | 9 | 4 | 4 | 0 | 0 | 4 | 0 |

d(1,5) = 4 + 0 = 4; d(2,5) = 12 + 0 = 12; d(3,6) = 8 + 4 = 12; d(4,1) = 4 + 9 = 13; d(5,2) = 0 + 4 = 4; d(5,4) = 0 + 0 = 0; d(6,3) = 0 + 4 = 4; d(6,4) = 0 + 0 = 0;  
Наибольшая сумма констант приведения равна (4 + 9) = 13 для ребра (4,1), следовательно, множество разбивается на два подмножества (4,1) и (4\*,1\*).  
**Исключение ребра** (4,1) проводим путем замены элемента d41 = 0 на M, после чего осуществляем очередное приведение матрицы расстояний для образовавшегося подмножества (4\*,1\*), в результате получим редуцированную матрицу.

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **i j** | **1** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | M | 4 | 4 | 8 | 0 | 47 | 0 |
| **2** | 12 | M | 12 | 24 | 0 | 28 | 0 |
| **3** | 9 | 8 | M | 37 | 41 | 0 | 0 |
| **4** | M | 26 | 41 | M | 34 | 4 | 4 |
| **5** | 19 | 0 | 25 | 0 | M | 9 | 0 |
| **6** | 31 | 4 | 0 | 0 | 33 | M | 0 |
| dj | 9 | 0 | 0 | 0 | 0 | 0 | 13 |

Нижняя граница гамильтоновых циклов этого подмножества:  
H(4\*,1\*) = 66 + 13 = 79  
**Включение ребра** (4,1) проводится путем исключения всех элементов 4-ой строки и 1-го столбца, в которой элемент d14 заменяем на М, для исключения образования негамильтонова цикла.  
В результате получим другую сокращенную матрицу (5 x 5), которая подлежит операции приведения.  
После операции приведения сокращенная матрица будет иметь вид:

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | 4 | 4 | M | 0 | 47 | 0 |
| **2** | M | 12 | 24 | 0 | 28 | 0 |
| **3** | 8 | M | 37 | 41 | 0 | 0 |
| **5** | 0 | 25 | 0 | M | 9 | 0 |
| **6** | 4 | 0 | 0 | 33 | M | 0 |
| dj | 0 | 0 | 0 | 0 | 0 | 0 |

Сумма констант приведения сокращенной матрицы:  
∑di + ∑dj = 0  
Нижняя граница подмножества (4,1) равна:  
H(4,1) = 66 + 0 = 66 ≤ 79  
Поскольку нижняя граница этого подмножества (4,1) меньше, чем подмножества (4\*,1\*), то ребро (4,1) включаем в маршрут с новой границей H = 66.  
**Шаг №2**.  
**Определяем ребро ветвления**.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | 4 | 4 | M | 0(4) | 47 | 4 |
| **2** | M | 12 | 24 | 0(12) | 28 | 12 |
| **3** | 8 | M | 37 | 41 | **0(17)** | 8 |
| **5** | 0(4) | 25 | 0(0) | M | 9 | 0 |
| **6** | 4 | 0(4) | 0(0) | 33 | M | 0 |
| dj | 4 | 4 | 0 | 0 | 9 | 0 |

d(1,5) = 4 + 0 = 4; d(2,5) = 12 + 0 = 12; d(3,6) = 8 + 9 = 17; d(5,2) = 0 + 4 = 4; d(5,4) = 0 + 0 = 0; d(6,3) = 0 + 4 = 4; d(6,4) = 0 + 0 = 0;  
max: d(3,6)=17.  
**Исключение ребра** (3,6): d36=M.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | **6** | di |
| **1** | 4 | 4 | M | 0 | 47 | 0 |
| **2** | M | 12 | 24 | 0 | 28 | 0 |
| **3** | 8 | M | 37 | 41 | M | 8 |
| **5** | 0 | 25 | 0 | M | 9 | 0 |
| **6** | 4 | 0 | 0 | 33 | M | 0 |
| dj | 0 | 0 | 0 | 0 | 9 | 17 |

H(3\*,6\*) = 66 + 17 = 83  
**Включение ребра** (3,6): d63=М.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | di |
| **1** | 4 | 4 | M | 0 | 0 |
| **2** | M | 12 | 24 | 0 | 0 |
| **5** | 0 | 25 | 0 | M | 0 |
| **6** | 4 | M | 0 | 33 | 0 |
| dj | 0 | 4 | 0 | 0 | 4 |

∑di + ∑dj = 4  
H(3,6) = 66 + 4 = 70 ≤ 83  
Ребро (3,6) включаем в маршрут с новой границей H=70.  
**Шаг №3**.  
**Определяем ребро ветвления**.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | di |
| **1** | 4 | **0(8)** | M | 0(0) | 0 |
| **2** | M | 8 | 24 | 0(8) | 8 |
| **5** | 0(4) | 21 | 0(0) | M | 0 |
| **6** | 4 | M | 0(4) | 33 | 4 |
| dj | 4 | 8 | 0 | 0 | 0 |

d(1,3) = 0 + 8 = 8; d(1,5) = 0 + 0 = 0; d(2,5) = 8 + 0 = 8; d(5,2) = 0 + 4 = 4; d(5,4) = 0 + 0 = 0; d(6,4) = 4 + 0 = 4;  
max: d(1,3)=8.  
**Исключение ребра** (1,3): d13=M.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **i j** | **2** | **3** | **4** | **5** | di |
| **1** | 4 | M | M | 0 | 0 |
| **2** | M | 8 | 24 | 0 | 0 |
| **5** | 0 | 21 | 0 | M | 0 |
| **6** | 4 | M | 0 | 33 | 0 |
| dj | 0 | 8 | 0 | 0 | 8 |

H(1\*,3\*) = 70 + 8 = 78  
**Включение ребра** (1,3): d31=М.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **2** | **4** | **5** | di |
| **2** | M | 24 | 0 | 0 |
| **5** | 0 | 0 | M | 0 |
| **6** | 4 | 0 | 33 | 0 |
| dj | 0 | 0 | 0 | 0 |

∑di + ∑dj = 0  
H(1,3) = 70 + 0 = 70 ≤ 78  
Запрещаем переходы: (6,4), (6,1),  
Ребро (1,3) включаем в маршрут с новой границей H=70.  
**Шаг №4**.  
**Определяем ребро ветвления**.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **2** | **4** | **5** | di |
| **2** | M | 24 | **0(57)** | 24 |
| **5** | 0(4) | 0(24) | M | 0 |
| **6** | 4 | M | 33 | 0 |
| dj | 4 | 24 | 33 | 0 |

d(2,5) = 24 + 33 = 57; d(5,2) = 0 + 4 = 4; d(5,4) = 0 + 24 = 24;  
max: d(2,5)=57.  
**Исключение ребра** (2,5): d25=M.

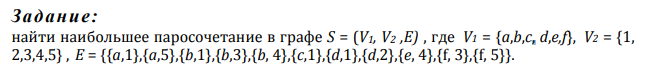
|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **i j** | **2** | **4** | **5** | di |
| **2** | M | 24 | M | 24 |
| **5** | 0 | 0 | M | 0 |
| **6** | 4 | M | 33 | 4 |
| dj | 0 | 0 | 33 | 61 |

H(2\*,5\*) = 70 + 61 = 131  
**Включение ребра** (2,5): d52=М.

|  |  |  |  |
| --- | --- | --- | --- |
| **i j** | **2** | **4** | di |
| **5** | M | 0 | 0 |
| **6** | 4 | M | 4 |
| dj | 4 | 0 | 8 |

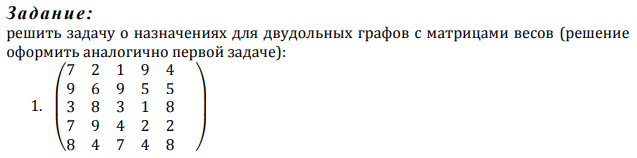
∑di + ∑dj = 8  
H(2,5) = 70 + 8 = 78 ≤ 131  
Ребро (2,5) включаем в маршрут с новой границей H=78.  
В соответствии с этой матрицей включаем в гамильтонов маршрут ребра (5,4) и (6,2).  
В результате по дереву ветвлений гамильтонов цикл образуют ребра:  
(4,1), (1,3), (3,6), (6,2), (2,5), (5,4),  
Длина маршрута равна F(Mk) = 74

# Задание 14 Паросочетания





C 1, D 2, B 3, E 4, A 5



**1. Проводим редукцию матрицы по строкам**. В связи с этим во вновь полученной матрице в каждой строке будет как минимум один ноль.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 6 | 1 | 0 | 8 | 3 | **1** |
| 4 | 1 | 4 | 0 | 0 | **5** |
| 2 | 7 | 2 | 0 | 7 | **1** |
| 5 | 7 | 2 | 0 | 0 | **2** |
| 4 | 0 | 3 | 0 | 4 | **4** |

Затем такую же операцию редукции проводим по столбцам, для чего в каждом столбце находим минимальный элемент.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1 | 0 | 8 | 3 |
| 2 | 1 | 4 | 0 | 0 |
| 0 | 7 | 2 | 0 | 7 |
| 3 | 7 | 2 | 0 | 0 |
| 2 | 0 | 3 | 0 | 4 |
| **2** | **0** | **0** | **0** | **0** |

После вычитания минимальных элементов получаем полностью редуцированную матрицу.  
**2. Методом проб и ошибок** проводим поиск допустимого решения, для которого все назначения имеют нулевую стоимость.  
Фиксируем нулевое значение в клетке (1, 3). Другие нули в строке 1 и столбце 3 вычеркиваем.  
Фиксируем нулевое значение в клетке (2, 4). Другие нули в строке 2 и столбце 4 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (3; 4), (4; 4), (5; 4), (2; 5).  
Фиксируем нулевое значение в клетке (3, 1). Другие нули в строке 3 и столбце 1 вычеркиваем.  
Фиксируем нулевое значение в клетке (4, 5). Другие нули в строке 4 и столбце 5 вычеркиваем.  
Фиксируем нулевое значение в клетке (5, 2). Другие нули в строке 5 и столбце 2 вычеркиваем.  
В итоге получаем следующую матрицу:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1 | **[0]** | 8 | 3 |
| 2 | 1 | 4 | **[0]** | [-0-] |
| **[0]** | 7 | 2 | [-0-] | 7 |
| 3 | 7 | 2 | [-0-] | **[0]** |
| 2 | **[0]** | 3 | [-0-] | 4 |

Количество найденных нулей равно k = 5. В результате получаем эквивалентную матрицу Сэ:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1 | 0 | 8 | 3 |
| 2 | 1 | 4 | 0 | 0 |
| 0 | 7 | 2 | 0 | 7 |
| 3 | 7 | 2 | 0 | 0 |
| 2 | 0 | 3 | 0 | 4 |

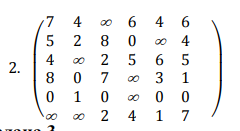
**4. Методом проб и ошибок определяем матрицу назначения Х**, которая позволяет по аналогично расположенным элементам исходной матрицы (в квадратах) вычислить минимальную стоимость назначения.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1 | **[0]** | 8 | 3 |
| 2 | 1 | 4 | [-0-] | **[0]** |
| **[0]** | 7 | 2 | [-0-] | 7 |
| 3 | 7 | 2 | **[0]** | [-0-] |
| 2 | **[0]** | 3 | [-0-] | 4 |

Cmin = 1 + 3 + 4 + 5 + 2 = 15  
Путь: (1;3), (3;1), (5;2), (2;5), (4;4)  
или цикл: (1;3), (3;1)  
Альтернативный вариант №2.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4 | 1 | **[0]** | 8 | 3 |
| 2 | 1 | 4 | **[0]** | [-0-] |
| **[0]** | 7 | 2 | [-0-] | 7 |
| 3 | 7 | 2 | [-0-] | **[0]** |
| 2 | **[0]** | 3 | [-0-] | 4 |

Cmin = 2 + 1 + 3 + 4 + 5 = 15



**1. Проводим редукцию матрицы по строкам**. В связи с этим во вновь полученной матрице в каждой строке будет как минимум один ноль.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 7 | 4 | 0 | 6 | 4 | 6 |  |
| 5 | 2 | 8 | 0 | 0 | 4 |  |
| 4 | 0 | 2 | 5 | 6 | 5 |  |
| 8 | 0 | 7 | 0 | 3 | 1 |  |
| 0 | 1 | 0 | 0 | 0 | 0 |  |
| 0 | 0 | 2 | 4 | 1 | 7 |  |

Затем такую же операцию редукции проводим по столбцам, для чего в каждом столбце находим минимальный элемент.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 7 | 4 | 0 | 6 | 4 | 6 |
| 5 | 2 | 8 | 0 | 0 | 4 |
| 4 | 0 | 2 | 5 | 6 | 5 |
| 8 | 0 | 7 | 0 | 3 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 2 | 4 | 1 | 7 |
| **0** | **0** | **0** | **0** | **0** | **0** |

После вычитания минимальных элементов получаем полностью редуцированную матрицу.  
**2. Методом проб и ошибок** проводим поиск допустимого решения, для которого все назначения имеют нулевую стоимость.  
Фиксируем нулевое значение в клетке (1, 3). Другие нули в строке 1 и столбце 3 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (5; 3)  
Фиксируем нулевое значение в клетке (2, 5). Другие нули в строке 2 и столбце 5 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (5; 5), (2; 4).  
Фиксируем нулевое значение в клетке (3, 2). Другие нули в строке 3 и столбце 2 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (4; 2), (6; 2)  
Фиксируем нулевое значение в клетке (4, 4). Другие нули в строке 4 и столбце 4 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (5; 4)  
Фиксируем нулевое значение в клетке (5, 6). Другие нули в строке 5 и столбце 6 вычеркиваем. Для данной клетки вычеркиваем нули в клетках (5; 1).  
Фиксируем нулевое значение в клетке (6, 1). Другие нули в строке 6 и столбце 1 вычеркиваем.  
В итоге получаем следующую матрицу:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 7 | 4 | **[0]** | 6 | 4 | 6 |
| 5 | 2 | 8 | [-0-] | **[0]** | 4 |
| 4 | **[0]** | 2 | 5 | 6 | 5 |
| 8 | [-0-] | 7 | **[0]** | 3 | 1 |
| [-0-] | 1 | [-0-] | [-0-] | [-0-] | **[0]** |
| **[0]** | [-0-] | 2 | 4 | 1 | 7 |

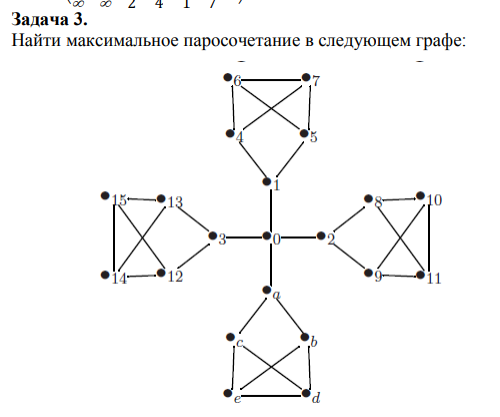
Количество найденных нулей равно k = 6. В результате получаем эквивалентную матрицу Сэ:

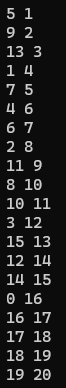
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 7 | 4 | 0 | 6 | 4 | 6 |
| 5 | 2 | 8 | 0 | 0 | 4 |
| 4 | 0 | 2 | 5 | 6 | 5 |
| 8 | 0 | 7 | 0 | 3 | 1 |
| 0 | 1 | 0 | 0 | 0 | 0 |
| 0 | 0 | 2 | 4 | 1 | 7 |

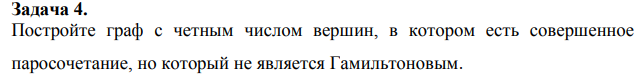
**4. Методом проб и ошибок определяем матрицу назначения Х**, которая позволяет по аналогично расположенным элементам исходной матрицы (в квадратах) вычислить минимальную стоимость назначения.

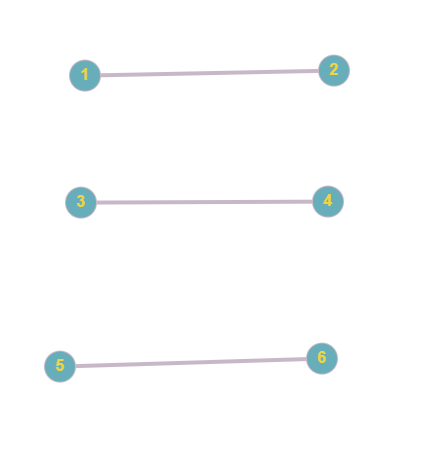
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 7 | 4 | **[0]** | 6 | 4 | 6 |
| 5 | 2 | 8 | [-0-] | **[0]** | 4 |
| 4 | **[0]** | 2 | 5 | 6 | 5 |
| 8 | [-0-] | 7 | **[0]** | 3 | 1 |
| [-0-] | 1 | [-0-] | [-0-] | [-0-] | **[0]** |
| **[0]** | [-0-] | 2 | 4 | 1 | 7 |

Cmin = + 0 = 0









# Задание 16 ИТОГ

1. Топ сорт total->ex\_1()
2. Поиск эйлерова пути
3. Поиск пути в покрашенном графе с переменной цветов
4. Поиск компонент связности total.h -> ex\_4()
5. Поиск мостов total.h -> ex\_55()
6. Поиск точек сочленения total.x -> ex\_66()
7. Конденсация графа
8. Дейкстра
9. Максимальный поток
10. Форд-беллман
11. Наибольшее паросочетание в двудольном графе Lesson15 -> ex()
12. Задача о назначении Программа workers (эвристикой)