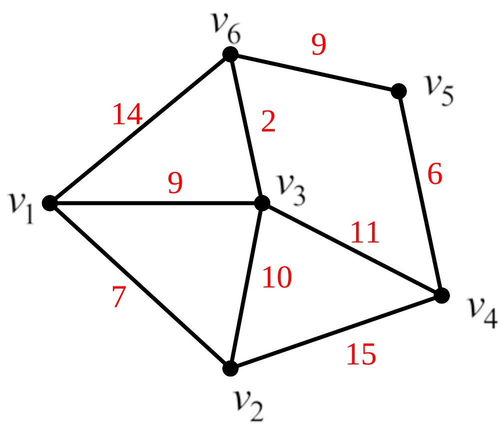
Тема 3.3. Поиск кратчайших маршрутов. Эйлеров и гамильтонов циклы и пути

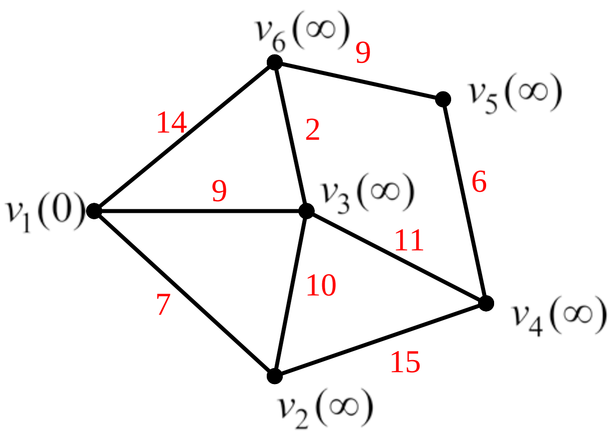
План: Задачи поиска маршрутов в графе. Диаметр, радиус и центр графа. Расстояния между вершинами графа. Эксцентриситет вершины. Диаметр графа. Центр графа. Обходы графов. Эйлеровы графы, эйлеровы цепи. Гамильтоновы графы.

Задачи с решениями

Пример 1: Пусть требуется найти кратчайшие расстояния от 1-й вершины до всех остальных.



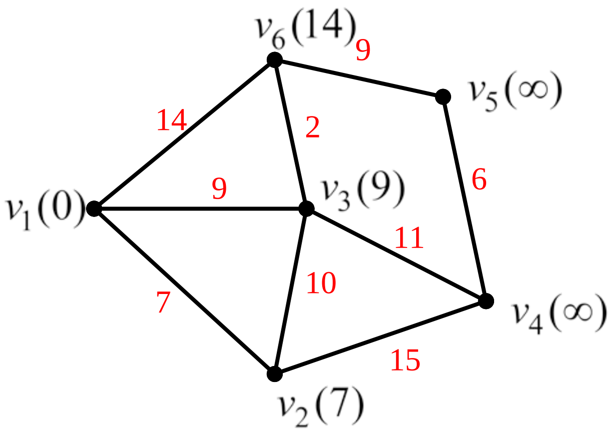
Инициализация: Метка самой вершины 1 полагается равной 0, метки остальных вершин – бесконечность (это вариант для составления программы для компьютера, для решения задачи на графах берут сразу расстояния до соседних вершин, т.е. то, что получится на следующем шаге).



Шаг 1. Соседи вершины с минимальной меткой (вершина v_1 с меткой 0) являются вершины 2, 3 и 6. Обходим соседей вершины по очереди.

Первый сосед v_1 – вершина v_2, потому что длина пути до неё минимальна. Длина пути $\rho(v_1, v_2)$ равна сумме кратчайшего расстояния до вершины v_1 (значению её метки, т.е. 0) и длины ребра, (v_1, v_2), то есть 0 + 7 = 7. Это меньше текущей метки v_2 (∞), поэтому новая метка 2-й вершины равна 7.

Аналогично находим длины пути для всех других соседей (вершины 3 и 6).



Все соседи v_1 проверены. Текущее минимальное расстояние до v_1 считается окончательным и пересмотру не подлежит. Вершина v_1 отмечается как посещенная (возьмем ее в квадратные скобки).

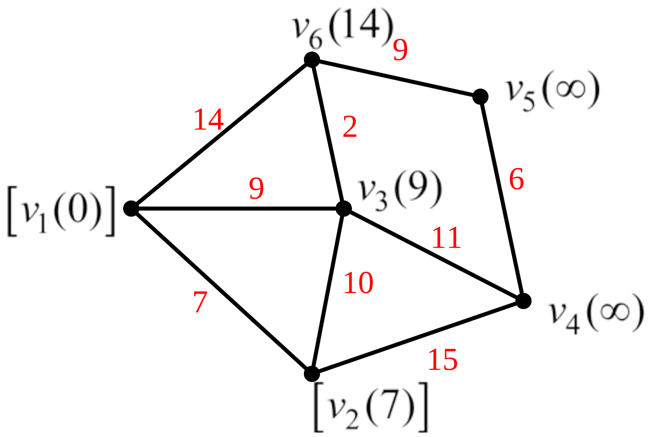
Шаг 2. Шаг 1 алгоритма повторяется. Снова находим «ближайшую» из непосещенных вершин. Это v_2 с меткой 7.

Снова пытаемся уменьшить метки соседей выбранной вершины, пытаясь пройти в них через v_2. Соседями вершины 2 являются вершины v_1, v_3 и v_4.

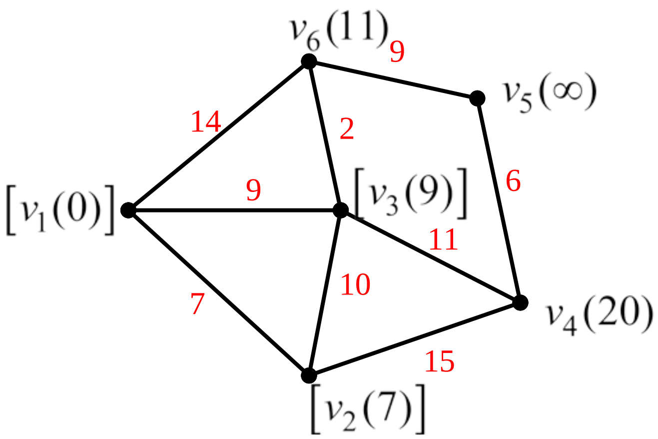
Вершина v_1 уже посещена. Следующий сосед вершины v_2 — вершина v_3, так как имеет минимальную метку из вершин, отмеченных как не посещённые. Если идти в неё через 2, то длина такого пути будет равна 17 (7 + 10 = 17). Но текущая метка третьей вершины равна 9, а 9 < 17, поэтому метка не меняется.

Ещё один сосед v_2 — вершина v_4. Если идти в неё через 2-ю, то длина такого пути будет равна 22 (7 + 15 = 22). Поскольку $22 < \infty$, устанавливаем метку v_4 равной 22.

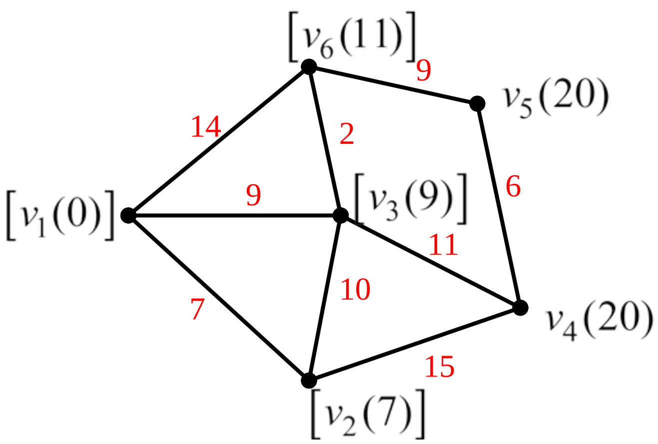
Все соседи вершины 2 просмотрены, помечаем её как посещенную.



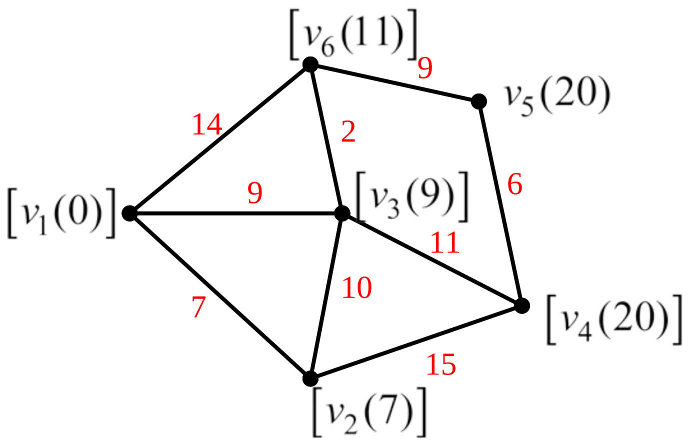
Шаг 3. Повторяем шаг алгоритма, выбрав вершину v_3. Получим:



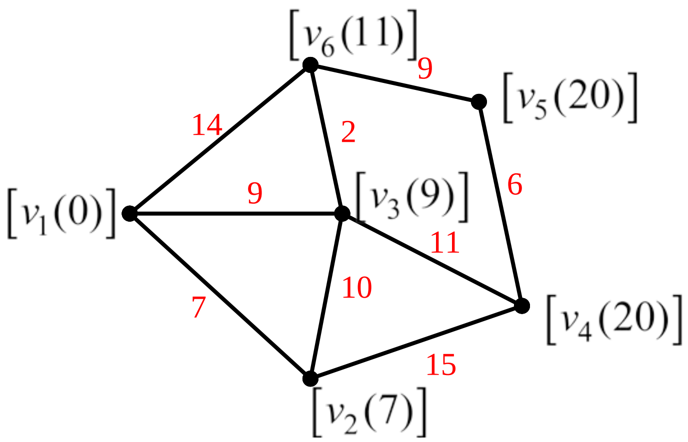
Шаг 4:



Шаг 5:



Шаг 6:



Таким образом, кратчайшие пути из вершины 1: (0,7,9,20,20,11).

Пример 2: По заданной матрице весов графа найти величину минимального пути от вершины x_1 до каждой из вершин по алгоритму Дейкстры (в матричном виде):

	x_1	x_2	x_3	x_4	x_5	x_6
x_1	0	11	∞	14	15	∞
x_2	∞	0	13	∞	∞	∞
x_3	∞	∞	0	∞	∞	15
x_4	8	7	11	0	9	∞
x_5	∞	12	10	∞	0	14
x_6	∞	∞	9	∞	∞	0

Решение:

Алгоритм Дейкстры является более эффективным, чем алгоритм Форда-Беллмана, но используется только для взвешенных графов, в которых веса всех дуг не отрицательны.

Матрица весов дана в условии.

Построим строку $T_1 = \{2,3,4,5,6\}$ - номера вершин до которых нужно вычислить длину пути и $D^{(1)} = (0,\underline{11},\infty,14,15,\infty)$ - расстояния от x_1 до этих вершин (первоначально совпадает с первой строкой матрицы весов). Находим минимальный элемент (подчеркнут) и удаляем его номер из строки Т. Пересчитываем D по правилу: $D^{(s)} = (d_1^{(s)},\dots d_n^{(s)})$, где $d_k^{(s+1)} = \min\left\{d_k^{(s)}, d_j^{(s)} + w_{jk}\right\}$, (т.е., если мы считаем *k*-ый элемент в строк *D*, то мы выбираем минимальное значение среди того элемента, который занимал эту позицию в предыдущей строке D, а также среди всех сумм элементов столбца с номером k матрицы весов и соответствующих, по порядку следования, значений предыдущей строки *D*) если $a_k \in T_{s+1}$, и $d_k^{(s+1)} = d_k^{(s)}$, если $a_k \notin T_{s+1}$. Получим:

$T_2 = \{3,4,5,6\}$.

$D^{(3)} = (0,11,24,\underline{14},15,29)$.

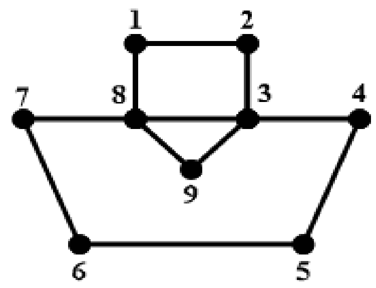
$T_3 = \{3,5,6\}$.

$D^{(4)} = (0,11,24,14,15,29)$.

Строка $D^{(4)}$ не отличается от $D^{(3)}$, поэтому решение закончено даже несмотря на то, что в строке Т остались элементы.

Ответ: минимальные расстояния от вершины 1 до всех остальных: $(0,11,24,14,15,29)$.

Пример: Для данного графа определить, есть ли в нем эйлеров цикл и, если есть, найти его.



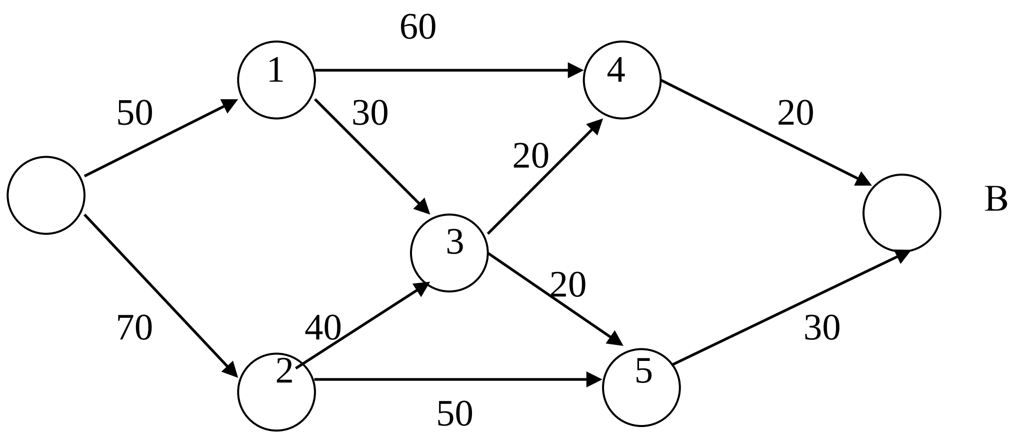
<u>Решение:</u>

Степени всех вершин графа четные, значит эйлеров цикл есть.

Начнем строить цикл с любой вершины, например построим: 1-2-3-9-8-1. Циклом охвачены не все ребра. Найдем вершину, уже включенную в наш цикл и которая также инцидентна ребрам, не включенным в цикл (например вершина 3). Из этой вершины построим цикл по ребрам, не вошедшим в цикл (3-4-5-6-7-8-3). А теперь объединим данные циклы «встроив» второй цикл в «вершину» 3 первого: 1-2-(3-4-5-6-7-8-3)-9-8-1. Получим: 1-2-3-4-5-6-7-8-3-9-8-1.

Задачи для самостоятельного решения

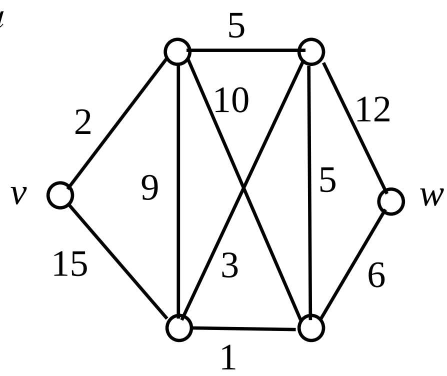
1. Автотранспортному предприятию предстоит освоить новый маршрут между городами *A* и *B*. На рисунке представлены различные маршруты следования из *A* в *B,* проходящие через несколько других поселков. Расстояния указаны (числами в километрах) около стрелок.



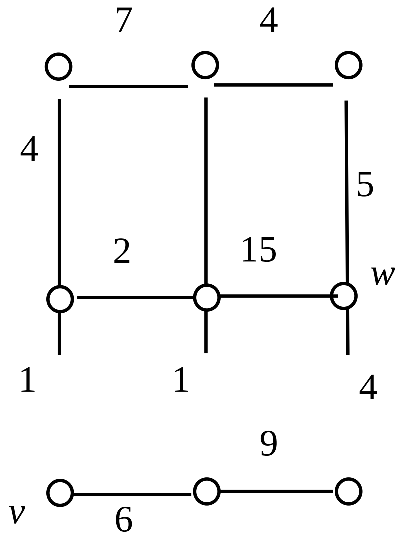
Требуется определить кратчайший маршрут следований автобусов из города *A* в город *B*.

2. Найти самый короткий путь из вершины v в вершину w в графах, показанных на рисунках (длина дуг приведена около соответствующих ребер).

а



б



3. Взвешенный граф имеет множество вершин (v_1, v_2, …, v_{10}), множество ребер (e_1, e_2, …, e_{16}) и матрицу инцидентности:

1	1	0	0	0	0	0	0	0	0
1	0	0	0	0	1	0	0	0	0
1	0	0	0	0	0	1	0	0	0
0	1	0	0	0	1	0	0	0	0

0	0	0	0	0	1	1	0	0	0
0	0	1	0	0	1	0	0	0	0
0	0	0	0	1	1	0	0	0	0
0	0	1	0	1	0	1	0	0	0
0	0	1	1	0	0	0	0	0	0
0	0	0	1	1	0	0	0	0	0
0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	1	0	0
0	0	0	0	0	0	0	1	1	0
0	0	0	0	0	1	0	0	1	0
0	0	0	0	0	0	0	0	1	1
0	0	0	0	0	0	1	0	0	1

Веса граней следующие:

Ребро	e_1	e_2	e_3	e_4	e_5	e_6	e_7	e_8	e_9	e_{10}	e_{11}	e_{12}	e_{13}	e_{14}	e_{15}	e_{16}
Вес	4	8	2	3	7	1	2	3	2	6	5	14	9	7	3	15

Нарисовать граф и найти кратчайший путь: а) из v_1 в v_2; б) из v_4 в v_{10}.

4. Определите, какой из графов 1–6 имеет эйлеров цикл, эйлеров путь? Гамильтонов цикл? Гамильтонов путь?