

ИПЗ ПО ДИСКРЕТНОЙ МАТЕМАТИКЕ

Тема: оптимизация на графах.

ТРЕБОВАНИЯ К ОФОРМЛЕНИЮ

- На титульном листе ставится номер варианта студента.
- Перед каждой задачей указывается номер задачи и в скобках – вариант данной задачи.
- Копируется из условия или переписывается условие задачи.
- Предоставляется развернутое и полное описание решения задачи.
- Выполненные задания
 - или фотографируются и фотографии в книжном формате последовательно помещаются в текстовый файл в формате Word после титульного листа;
 - или набираются в текстовом редакторе после титульного листа.
 - при использовании программы Excel представить скриншоты шагов решения с комментариями.

Отдельные фотографии к проверке не принимаются!

Работа представляется не позднее, чем за неделю до экзамена!!!

Методические указания

Тема: задача коммивояжера.

Описание алгоритма.

Коммивояжер – разъездной торговый агент какой-либо фирмы, предлагающий покупателям товары по образцам и каталогам.

Одна из известных и важных задач оптимизации (а также логистики) - задача коммивояжера (англ. «Travelling salesman problem», TSP). В некоторой литературе встречается название «задача о бродячем торговце».

Суть задачи - необходимо определить кратчайший путь (оптимальный), проходящий через некие пункты по одному разу.

Например, данная задача может применяться для нахождения самого выгодного маршрута, позволяющего коммивояжеру объехать определенные города со своим товаром по одному разу и вернуться в исходную точку.

Мерой выгодности маршрута будет минимальное время, проведенное в пути, минимальные расходы на дорогу или, в самом простом случае, минимальная длина пути

Кто и когда впервые начал исследовать задачу коммивояжера неизвестно, но одним из первых предложил решение подобной проблемы выдающийся математик XIX в.

Рассмотрим замкнутый вариант задачи (в результате решения возвращаемся в исходную точку) с помощью **метода ветвей и границ**.

(метод ветвей и границ — один из методов дискретной оптимизации, являющийся развитием метода полного перебора, но отличающийся от него отсевом в процессе вычисления подмножеств неэффективных решений. Впервые был предложен в 1960 году английским профессором Алисой Лэнд и австралийским математиком Элисон Дойг.).

Общий план решения.

1. **Построение матрицы с исходными данными** — в таблицу заносятся расстояния (C_{ij}) между городами (в ячейки типа А-А, В-В и т. д. ставится символ М — условно бесконечно большое число); при этом строкам соответствуют города отбытия, а столбцам города прибытия.
2. **Нахождение минимума по строкам** — в каждой строке определяется минимальное число (d_i) и выписывается в отдельный столбец
3. **Редукция строк** — из значений ячеек каждой строки вычитаем соответствующий минимум ($C_{ij} = C_{ij} - d_i$), не затрагивая при этом клетки с М;
4. **Нахождение минимума по столбцам** — в каждом столбце определяется минимальное число (d_j) и выписывается в отдельную строку.
5. **Редукция столбцов** — из значений ячеек каждого столбца вычитаем соответствующий минимум ($C_{ij} = C_{ij} - d_j$), не затрагивая при этом клетки с М.
6. **Нахождение корневой нижней границы** (делаем это только один раз, в следующие разы пункт 6 пропускаем) — вычисляем нижнюю границу (минимально возможную на текущем этапе длину маршрута) в стартовой (корневой) точке решения, как сумму найденных ранее минимумов ($H_0 =$

$\sum d_i + \sum d_j$) и начинаем построение графа (схемы) решения с внесения в него корневой вершины.

7. **Вычисление оценок нулевых клеток** — считаем оценки (p_{ij}) для каждой ячейки с нулями, как сумму минимумов по строке и столбцу, в которых располагается нулевая клетка, не учитывая при этом саму нулевую клетку.
8. **Выбор нулевой клетки с максимальной оценкой** — ищем среди нулевых клеток обладающую наибольшей оценкой (если таких ячеек несколько, выбираем любую), и получаем пару ветвей (вариантов) решения задачи: с включением в маршрут отрезка пути относящегося к выбранной ячейке и без включения
9. **Редукция матрицы** — вычеркиваем относящиеся к выбранной клетке строку и столбец, а также заменяем значение ячейки соответствующей обратному пути на M .
10. **Вычисление нижней границы первой ветви** (включающей отрезок пути) — вновь находим минимумы по строкам, проводим редукцию строк, находим минимумы по столбцам, проводим редукцию столбцов, после чего вычисляем локальную нижнюю границу, как сумму предыдущей локальной нижней границы и минимумов ($H_k = H_{k-1} + \sum d_i + \sum d_j$), и добавляем вершину в граф.
11. **Вычисление нижней границы второй ветви** (не включающей отрезок пути) — считаем локальную нижнюю границу, как сумму предыдущей локальной нижней границы и оценки выбранной ранее нулевой клетки ($H_k^* = H_{k-1} + p_{ij}$), и добавляем вершину в граф.
12. **Выбор ветви с минимальным значением нижней границы** — среди еще не ветвившихся вершин выбираем обладающую минимальным значением локальной нижней границы (вне зависимости от того, какую ветвь рассматриваем в данный момент).
13. **Если полный маршрут еще не найден, продолжаем решение, если найден — переходим к пункту 10** — если маршрут еще не найден, то ход дальнейшего решения зависит от выбранной ветви: (а) первая ветвь — переходим к пункту 7, (б) вторая ветвь — в клетку с максимальной оценкой ставим M и переходим к пункту 2, (в) другая ветвь — возвращаемся к соответствующим ей этапу решения и таблице данных.
14. **Построение полного маршрута и определение его длины** — соединяем все найденные ранее отрезки пути в полный маршрут и считаем его общую длину (данные берем из исходной таблицы).

Пример.

1. Построение матрицы с исходными данными.

Город	A	B	C	D	E
A	M	20	18	12	8
B	5	M	14	7	11
C	12	18	M	6	11
D	11	17	11	M	12
E	5	5	5	5	M

В рассматриваемом примере у нас 5 городов (обозначенных буквами латинского алфавита) и в таблице указано расстояние от каждого города к 4-м другим. Названия строк соответствуют городам отправления, названия столбцов — городам назначения. В ячейках, расположенных на пересечениях строк и столбцов, указано расстояние между соответствующими городами.

Обратите внимание, что в зависимости от направления движения расстояние между городами отличается (конечно в реальности расстояние при движении от пункта А к пункту В будет скорее всего одинаковым, но не обязательно: к примеру, пункты могут соединять дороги с односторонним движением различной длины; или, если мы говорим о стоимости, билеты на прямой и обратный рейс могут иметь разную цену).

Расстояние от города к этому же городу обозначено буквой М. Также используется знак бесконечности (∞) или сокращение «Inf» (от англ. «Infinity» — бесконечность). Это означает, что данный отрезок пути условно принят за бесконечно длинный. Тогда при решении задачи не будет математического смысла выбирать движение от 1-ого города к 1-му, от 2-ого ко 2-му, и т. п. в качестве отрезка итогового маршрута.

2. Нахождение минимумов по строкам

Находим минимальное значение в каждой строке (d_i) и выписываем его в отдельный столбец.

Город	A	B	C	D	E	d_i
A	M	20	18	12	8	8
B	5	M	14	7	11	5
C	12	18	M	6	11	6
D	11	17	11	M	12	11
E	5	5	5	5	M	5

Найденные значение d_i называются **константами приведения для строк**.

3. Редукция строк

Производим редукцию строк – из каждого элемента в каждой строке вычитаем соответствующее ей значение минимума ($C_{ij} = C_{ij} - d_i$).

Город	A	B	C	D	E	d_i
A	M	12	10	4	0	8
B	0	M	9	2	6	5
C	6	12	M	0	5	6
D	0	6	0	M	1	11
E	0	0	0	0	M	5

В итоге получаем таблицу с новыми данными, в каждой строке которой будет хотя бы одна нулевая клетка.

4. Нахождение минимумов по столбцам

Далее находим минимальные значения в каждом столбце (d_j). Эти минимумы (включая нули) выписываем в отдельную строку.

Город	A	B	C	D	E
A	M	12	10	4	0
B	0	M	9	2	6
C	6	12	M	0	5
D	0	6	0	M	1
E	0	0	0	0	M
d_j	0	0	0	0	0

Найденные значение d_j называются **константами приведения для столбцов**.

Так совпало, что здесь все они нулевые, но конечно их значения могут быть и больше нуля.

5. Редукция столбцов

Производим редукцию столбцов — вычитаем из каждого элемента каждого столбца соответствующее ему значение минимума ($C_{ij} = C_{ij} - d_j$).

Город	A	B	C	D	E
A	M	12	10	4	0
B	0	M	9	2	6
C	6	12	M	0	5
D	0	6	0	M	1
E	0	0	0	0	M
d_j	0	0	0	0	0

В итоге в каждом столбце будет хотя бы одна нулевая клетка.

6. Нахождение корневой нижней границы

На этом этапе следует провести небольшое, но крайне важное вычисление, а именно определить корневую локальную нижнюю границу (H_k) длины (стоимости, длительности) маршрута. Для этого нужно суммировать константы приведения d_i и d_j .

Обратите внимание, что мы не вычисляем заново константы приведения для строк, а используем найденные ранее значения.

Город	A	B	C	D	E	d_i
A	M	12	10	4	0	8
B	0	M	9	2	6	5
C	6	12	M	0	5	6
D	0	6	0	M	1	11
E	0	0	0	0	M	5
d_j	0	0	0	0	0	

Формула для вычисления локальной нижней границы в корневой (стартовой) точке решения следующая (далее мы будем использовать несколько отличающиеся формулы): $H_0 = \sum d_i + \sum d_j$.

$$H_0 = (8 + 5 + 6 + 11 + 5) + (0 + 0 + 0 + 0 + 0) = 35 + 0 = 35.$$

Найденное значение **$H = 35$** является **текущей или локальной нижней границей**.

**Корневую локальную нижнюю границу считаем только единожды!
В последующие разы этот пункт пропускаем.**

Ход решения задачи коммивояжера удобно представить в виде графа, где вершинами будут ключевые решения по включению / невключению в итоговый маршрут тех или иных отрезков пути, а ребра, показывающие ход решения, образуют ветви альтернативных вариантов маршрута. Пока у нас есть только одна вершина — «корень» дерева решения, куда мы вписываем значение локальной нижней границы $H = 35$.

Корень дерева
решения
35

По сути значение H , это ни что иное как текущая минимальная длина маршрута. Т. е. короче вычисляемый маршрут быть не может, длиннее — возможно.

7. Вычисление оценок нулевых клеток

Для каждой нулевой клетки преобразованной матрицы находим «оценку» (r_{ij}). Ею будет сумма минимума по строке и минимума по столбцу, на пересечении которых находится данная клетка с нулем. При этом сама нулевая клетка для которой вычисляется оценка и найденные ранее d_i и d_j НЕ учитываются (но другие нулевые клетки, если они есть в рассматриваемых строке и столбце, учитывать нужно; также учитываются ячейки с M , которые означают бесконечно большое число). Полученную оценку записываем рядом с нулем, в скобках.

Например, первая строка таблицы содержит одну нулевую клетку А-Е. Вычислим для нее оценку: 4 (минимальное значение среди ячеек строки) + 1 (минимальное значение среди ячеек столбца) = 5.

Город	А	В	С	Д	Е
А	М	12	10	4	0 (5)
В	0	М	9	2	6
С	6	12	М	0	5
Д	0	6	0	М	1
Е	0	0	0	0	М

Аналогично находим оценки для всех остальных нулевых клеток:

Город	A	B	C	D	E
A	M	12	10	4	0 (5)
B	0 (2)	M	9	2	6
C	6	12	M	0 (5)	5
D	0 (0)	6	0 (0)	M	1
E	0 (0)	0 (6)	0 (0)	0 (0)	M

Вычисленные нами оценки также называются **штрафами** за неиспользование тех отрезков маршрута, которым соответствуют клетки с нулями (напоминаю, что каждая клетка «указывает» на два города: по своей строке — на город из которого выезжаем, и по своему столбцу — на город, в который приезжаем).

Штраф указывают на затраты расстояния, времени, стоимости (или иного параметра, в целях оптимизации которого мы решаем задачу коммивояжера), которые появятся, если НЕ выбрать данную клетку.

8. Выбор нулевой клетки с максимальной оценкой

Следующий важный шаг — выбор среди нулевых клеток матрицы той, что имеет наибольшую оценку. Это делается для того, чтобы избежать максимального удлинения маршрута, которое появится если НЕ выбрать такую ячейку.

Город	A	B	C	D	E
A	M	12	10	4	0 (5)
B	0 (2)	M	9	2	6
C	6	12	M	0 (5)	5
D	0 (0)	6	0 (0)	M	1
E	0 (0)	0 (6)	0 (0)	0 (0)	M

В нашем случае максимальную оценку имеет клетку E-B. Этой ячейке соответствует отрезок пути от города E к городу B.

Стоит отметить, что, когда не одна, а несколько нулевых ячеек имеют одинаковую максимальную оценку, можно выбрать любую из них.

9. Редукция матрицы.

На этом этапе решение задачи начинает **ветвиться**. Возникает развилка с парой альтернативных вариантов:

- решения, где мы включаем в маршрут выбранный отрезок пути (Е-В);
- и ветвь, где мы его в маршрут НЕ включаем.

Для начала рассмотрим *первую ветвь* решения, предусматривающую включение отрезка в маршрут. Для проверки этого варианта необходимо провести **редукцию матрицы**. Делается это следующим образом: **выбрав клетку с максимальной оценкой, вписываем в нее букву М, и затем полностью вычеркиваем соответствующие ей строку и столбец.**

В нашем примере мы включаем в итоговый маршрут движение из города Е в город В. Следовательно, из города Е в другие города выезжать уже не будем, а в город В по условиям задачи дважды заезжать нельзя. Поэтому данные строка и столбец нам более не понадобятся.

Город	А	В	С	Д	Е
А	М	12	10	4	0
В	0	М	9	2	6
С	6	12	М	0	5
Д	0	6	0	М	1
Е	0	М	0	0	М

Важный момент! В клетку, соответствующую обратному пути (В-Е), ставим еще одну М (т. к. мы уже не будем возвращаться обратно из города В в город Е).

Город	А	С	Д	Е
А	М	10	4	0
В	0	9	2	М
С	6	М	0	5
Д	0	0	М	1

В результате редукции (исключения строки и столбца нулевой клетки с максимальной оценкой) получаем новую уменьшенную таблицу.

10. Вычисление нижней границы первой ветви

Теперь необходимо вычислить **локальную нижнюю границу** для первой ветви решения задачи, предполагающей **включение** в итоговый маршрут выбранного отрезка пути (у нас это Е-В).

Чтобы это сделать, для новой таблицы вновь находим минимумы по строкам и проводим редукцию строк.

Город	A	C	D	E	d_i
A	M	10	4	0	0
B	0	9	2	M	0
C	6	M	0	5	0
D	0	0	M	1	0

Далее находим минимумы по столбцам и проводим редукцию столбцов.

Город	A	C	D	E
A	M	10	4	0
B	0	9	2	M
C	6	M	0	5
D	0	0	M	1
d_j	0	0	0	0

Затем вычисляем значение **локальной нижней границы** по формуле:

$$H_k = H_{k-1} + \sum d_i + \sum d_j.$$

Как видите, в этом случае локальная нижняя граница рассчитывается немного не так, как в пункте 6. Здесь это сумма предыдущей локальной нижней границы ветвящейся вершины (т. е. того участка графа, из которого исходят рассматриваемые ветви) и констант приведения:

$$H_1 = 35 + (0 + 0 + 0 + 0) + (0 + 0 + 0 + 0) = 35.$$

Полученное значение — локальная нижняя граница для данной ветви решения. Оно означает, что включение отрезка пути Е-В на данном этапе не приводит к удлинению итогового маршрута (поскольку корневая нижняя граница также равна 35).

11. Вычисление нижней границы для второй ветви

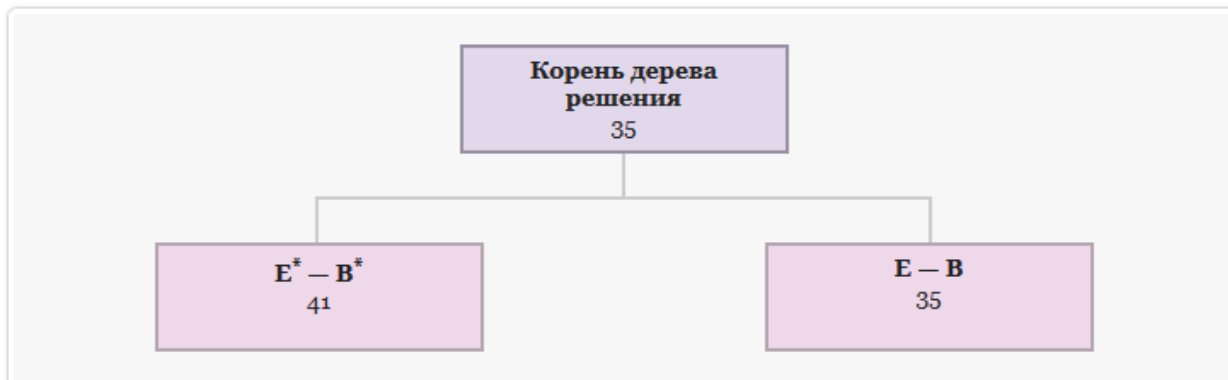
Далее необходимо проверить **вторую ветвь** — вычислим для нее значение локальной нижней границы. В этом случае мы НЕ исключаем из таблицы отрезок Е-В и локальная нижняя граница будет равна сумме предыдущей локальной нижней границы и максимальной оценки (т. е. оценки той нулевой клетки, которую мы выбрали на предыдущем шаге):

$$H_k^* = H_{k-1} + p_{ij}.$$

Т. о. локальная нижняя граница для второй ветви (позначим ее звездочкой) составит $H_1^* = 35 + 6 = 41$. Как видите, данная ветвь решения увеличивает минимально возможную длину итогового маршрута до значения 41.

12. Выбор ветви с минимальным значением нижней границы

Прежде всего, для наглядности, добавим обе рассмотренных ветви решения задачи в **граф**. Здесь рассматриваемая пара ветвей исходит из корневой вершины. Запишем для каждой ветви относящийся к ней отрезок пути (звездочками помечена ветвь, в которой отрезок пути НЕ включен в итоговый маршрут) и значение локальной нижней границы.



Теперь из всех (!) НЕ ветвившихся вершин графа выбираем ту, что имеет **наименьшее** значение локальной нижней границы.

Еще раз обращаю ваше внимание, что мы учитываем только те вершины (прямоугольники на рисунке), которые еще НЕ ветвились. К примеру, на данном этапе корень дерева решения уже ветвился, и мы его не учитываем, поэтому выбор ведется среди двух ветвей, заканчивающихся вершинами E*-B* и E-B.

Важный нюанс — при выборе следует рассматривать не только проверяемую в текущий момент пару ветвей, но и все другие ветви, даже отринутые на предыдущих этапах (пока таких нет, но далее по ходу решения задачи подобный случай будет разобран подробнее).

Поскольку в нашем примере пока всего две ветви и та, что включает в маршрут отрезок $E \rightarrow B$, имеет значение локальной нижней границы меньше, чем в случае альтернативного варианта ($35 < 41$), выбираем ее.

13. Если полный маршрут еще не найден, продолжаем решение задачи, если найден — переходим к пункту 14

Если мы еще не нашли все отрезки пути, то продолжаем решение задачи и здесь возможны 3 варианта:

- выбор ветви, включающей рассматриваемый отрезок — в этом случае решение задачи продолжается с пункта 7. Вновь находить минимумы по строкам и столбцам, а также проводить редукцию строк и столбцов не нужно, т. к. все это уже было сделано при вычислении локальной нижней границы в пункте 10. Поэтому сразу переходим к этапу вычисления оценок нулевых клеток;
- выбор ветви, НЕ включающей рассматриваемый отрезок — такой вариант предусматривает исключение из итогового маршрута искомого отрезка, для чего в соответствующую ему клетку таблицы

необходимо поставить М. Затем возвращаемся к пункту 2 и продолжаем решение задачи;

- выбор другой ветви — здесь мы возвращаемся к соответствующим этой ветви этапу решения и таблице с данными.

В нашем случае мы выбрали вариант с включением в итоговый маршрут ветви содержащей отрезок пути Е-В, а значит вновь находить минимумы по строкам и столбцам, а также проводить их редукцию не нужно — это мы уже делали ранее. Поэтому сразу вычисляем оценки для нулевых клеток и определяем ячейку с максимальной (D-C).

Город	A	C	D	E
A	M	10	4	0 ⁽⁵⁾
B	0 ⁽²⁾	9	2	M
C	6	M	0 ⁽⁷⁾	5
D	0 ⁽⁰⁾	0 ⁽⁹⁾	M	1

Рассматриваем два варианта: с включением ветви $D \rightarrow C$ в маршрут и без этой ветви. В последнем случае, как мы подсчитали выше, локальная минимальная нижняя граница будет равна $H^* = 35$ (предыдущая локальная нижняя граница) + 9 (максимальная оценка нулевой клетки) = 44.

Рассмотрим теперь первый вариант, предусматривающий включение ветви D-C в маршрут. Для этого проводим редукцию матрицы и ставим М на обратный путь (C-D).

Город	A	D	E
A	M	4	0
B	0	2	M
C	6	M	5

Вычисляем минимумы по строкам.

Город	A	D	E	d_i
A	M	4	0	0
B	0	2	M	0
C	6	M	5	5

Проводим редукцию строк.

Город	A	D	E
A	M	4	0
B	0	2	M
C	1	M	0
d_j	0	2	0

Находим минимумы по столбцам.

Город	A	D	E
A	M	4	0
B	0	2	M
C	1	M	0
d_j	0	2	0

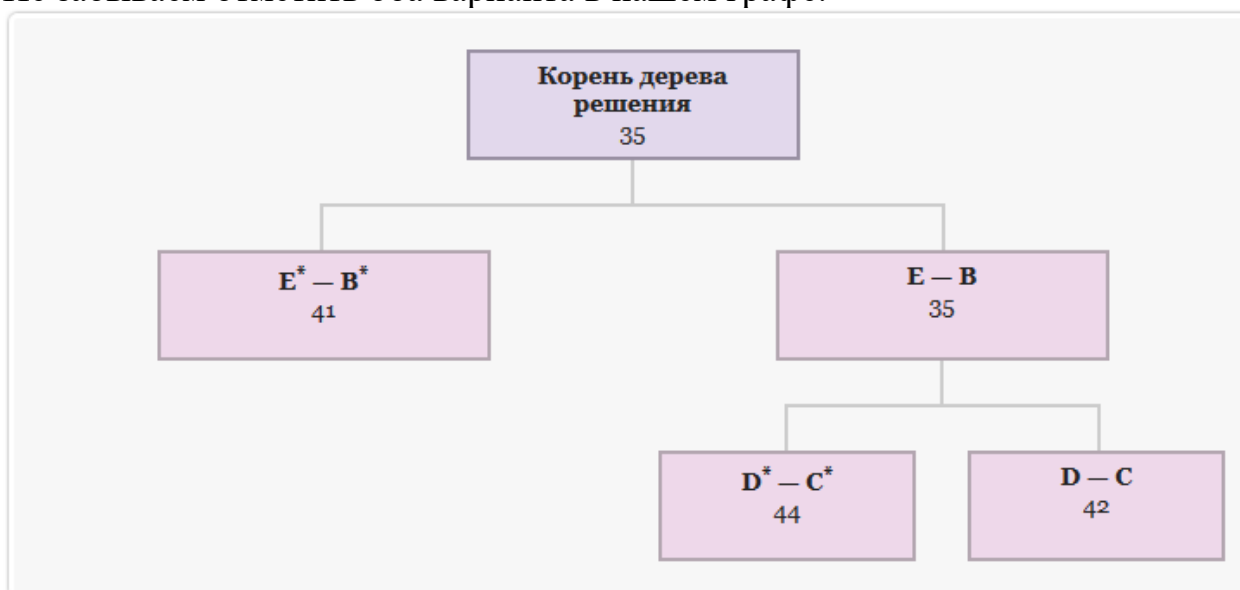
Проводим редукцию столбцов.

Город	A	D	E
A	M	2	0
B	0	0	M
C	1	M	0
d_j	0	2	0

Определяем локальную нижнюю границу:

$$H = 35 + (0 + 0 + 5) + (0 + 2 + 0) = 42.$$

Не забываем отметить оба варианта в нашем графе.



Теперь выбираем из всех еще не ветвившихся вершин ту, что имеет наименьшее значение локальной нижней границы. Здесь этому условию отвечает вершина $E^* — B^*$, которая имеет локальную нижнюю минимальную границу равную 41. Поэтому нам необходимо вернуться и продолжить развитие этой ветви решения задачи.

Если мы отбрасываем одну из ветвей и возвращаемся обратно, нам необходимо эту ветвь полностью исключить для дальнейшего рассмотрения, поэтому в соответствующую ей ячейку нужно поставить М, а также заново определить константы приведения и провести новую редукцию строк и столбцов!

Итак, возвращаемся по графу к ветви, НЕ предусматривающей включение отрезка Е-В в итоговый маршрут. Поскольку, данный отрезок пути был нами отвергнут, необходимо поставить в ячейку Е-В символ М.

Город	А	В	С	Д	Е
А	М	12	10	4	0
В	0	М	9	2	6
С	6	12	М	0	5
Д	0	6	0	М	1
Е	0	М	0	0	М

Определяем минимумы по строкам.

Город	А	В	С	Д	Е	d_i
А	М	12	10	4	0	0
В	0	М	9	2	6	0
С	6	12	М	0	5	0
Д	0	6	0	М	1	0
Е	0	М	0	0	М	0

Проводим редукцию строк. Т. к. минимумы или иначе константы приведения для строк (d_i) нулевые, данные не изменятся.

Город	A	B	C	D	E	d_i
A	M	12	10	4	0	0
B	0	M	9	2	6	0
C	6	12	M	0	5	0
D	0	6	0	M	1	0
E	0	M	0	0	M	0

Определяем минимумы по столбцам.

Город	A	B	C	D	E
A	M	12	10	4	0
B	0	M	9	2	6
C	6	12	M	0	5
D	0	6	0	M	1
E	0	M	0	0	M
d_j	0	6	0	0	0

Проводим редукцию столбцов.

Город	A	B	C	D	E
A	M	6	10	4	0
B	0	M	9	2	6
C	6	6	M	0	5
D	0	0	0	M	1
E	0	M	0	0	M
d_j	0	6	0	0	0

Вычисляем оценки нулевых клеток преобразованной матрицы и выбираем среди них максимальную.

Город	A	B	C	D	E
A	M	6	10	4	0 ⁽⁵⁾
B	0 ⁽²⁾	M	9	2	6
C	6	6	M	0 ⁽⁵⁾	5
D	0 ⁽⁰⁾	0 ⁽⁶⁾	0 ⁽⁰⁾	M	1
E	0 ⁽⁰⁾	M	0 ⁽⁰⁾	0 ⁽⁰⁾	M

Здесь клетка с максимальной оценкой D-B. Проводим редукцию матрицы, вычеркивая соответствующие этой ячейке строку и столбец. Не забываем поставить на обратный путь (B-D) букву М.

Город	A	C	D	E
A	M	10	4	0
B	0	9	M	6
C	6	M	0	5
E	0	0	0	M

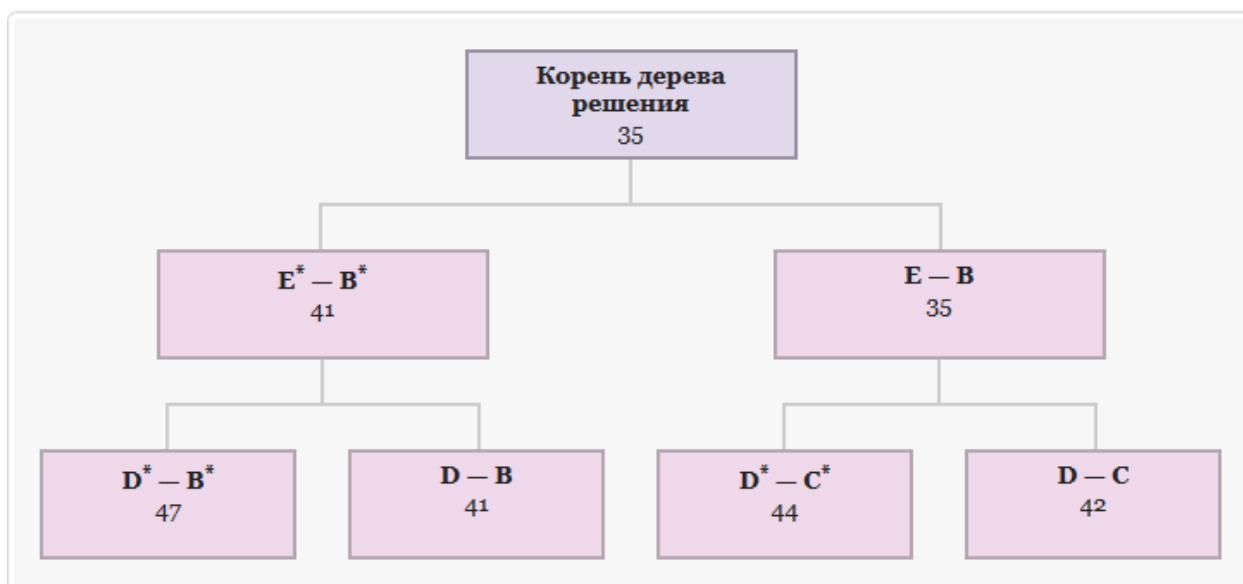
Найдем константы приведения, а также проведем редукцию строк и столбцов для новой таблицы.

Город	A	C	D	E	d_i
A	M	10	4	0	0
B	0	9	M	6	0
C	6	M	0	5	0
E	0	0	0	M	0
d_j	0	0	0	0	

Здесь минимумы по строкам и столбцам равны 0, а, следовательно, данные не меняются. Далее определим локальную нижнюю границу:

$$H = 41 + (0 + 0 + 0 + 0) + (0 + 0 + 0 + 0) = 41.$$

Что касается варианта, не предусматривающего включение отрезка D-B в итоговый маршрут, то для него $H^* = 41 + 6 = 47$. Занесем все эти данные в граф.



Поскольку 41 меньше, чем локальные нижние границы других ветвей, выбираем данную ветвь, предусматривающую включение в итоговый маршрут отрезка D-B.

Продолжаем решение задачи. Теперь следует вычислить оценки нулевых клеток и определить среди них максимальную.

Город	A	C	D	E
A	М	10	4	0 (9)
B	0 (6)	9	М	6
C	6	М	0 (5)	5
E	0 (0)	0 (9)	0 (0)	М

Вновь у нас две клетки с одинаковой максимальной оценкой равной 9: А-Е и Е-С. Выбираем любую, например А-Е.

Проверяем новые ветви: с включением в итоговый маршрут отрезка А-Е и без этого отрезка пути.

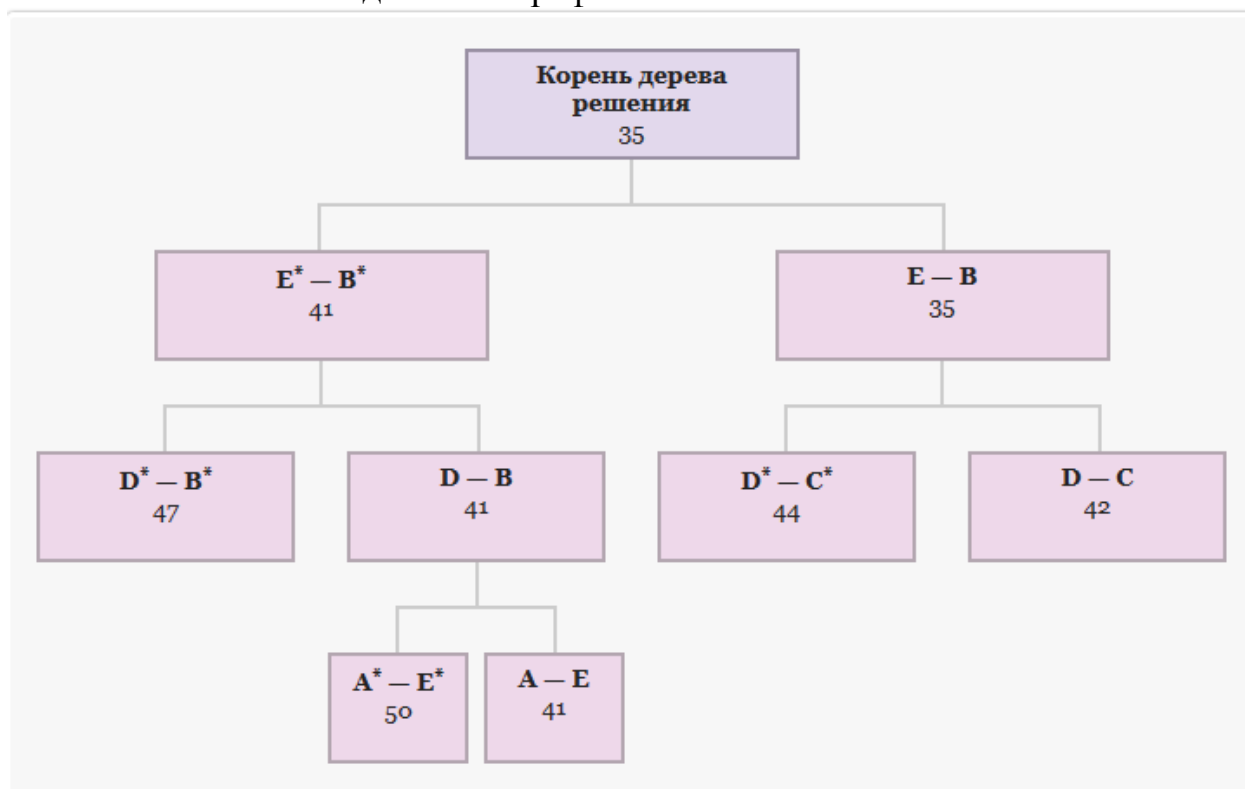
Для варианта с включением отрезка в маршрут проводим редукцию матрицы, а также исключаем обратный путь (Е-А).

Город	A	C	D
B	0	9	М
C	6	М	0
E	М	0	0

Находим минимумы по строкам — везде 0, проводим редукцию строк (данные, понятное дело, в этом случае не изменятся). Находим минимумы по столбцам — также 0, проводим редукцию столбцов (данные остаются прежними).

Считаем $H = 41 + (0 + 0 + 0) + (0 + 0 + 0) = 41$.

Для ветви, не включающей отрезок пути А-Е, значение $H^* = 41 + 9 = 50$.
Занесем все эти данные в граф.



Т. к. нет других ветвей решения со значением локальной нижней границы меньше 41, выбираем ветвь с включением отрезка пути А-Е в итоговый маршрут и продолжаем ее решение.

Переходим к этапу вычисления оценок нулевых клеток и определения максимальной среди них.

Город	А	С	Д
В	0 (15)	9	М
С	6	М	0 (6)
Е	М	0 (9)	0 (0)

Ячейка В-А имеет максимальную оценку 15, поэтому выбираем ее. Проводим редукцию матрицы. Исключать обратный путь (А-В) нужды нет, поскольку такая ячейка в таблице отсутствует.

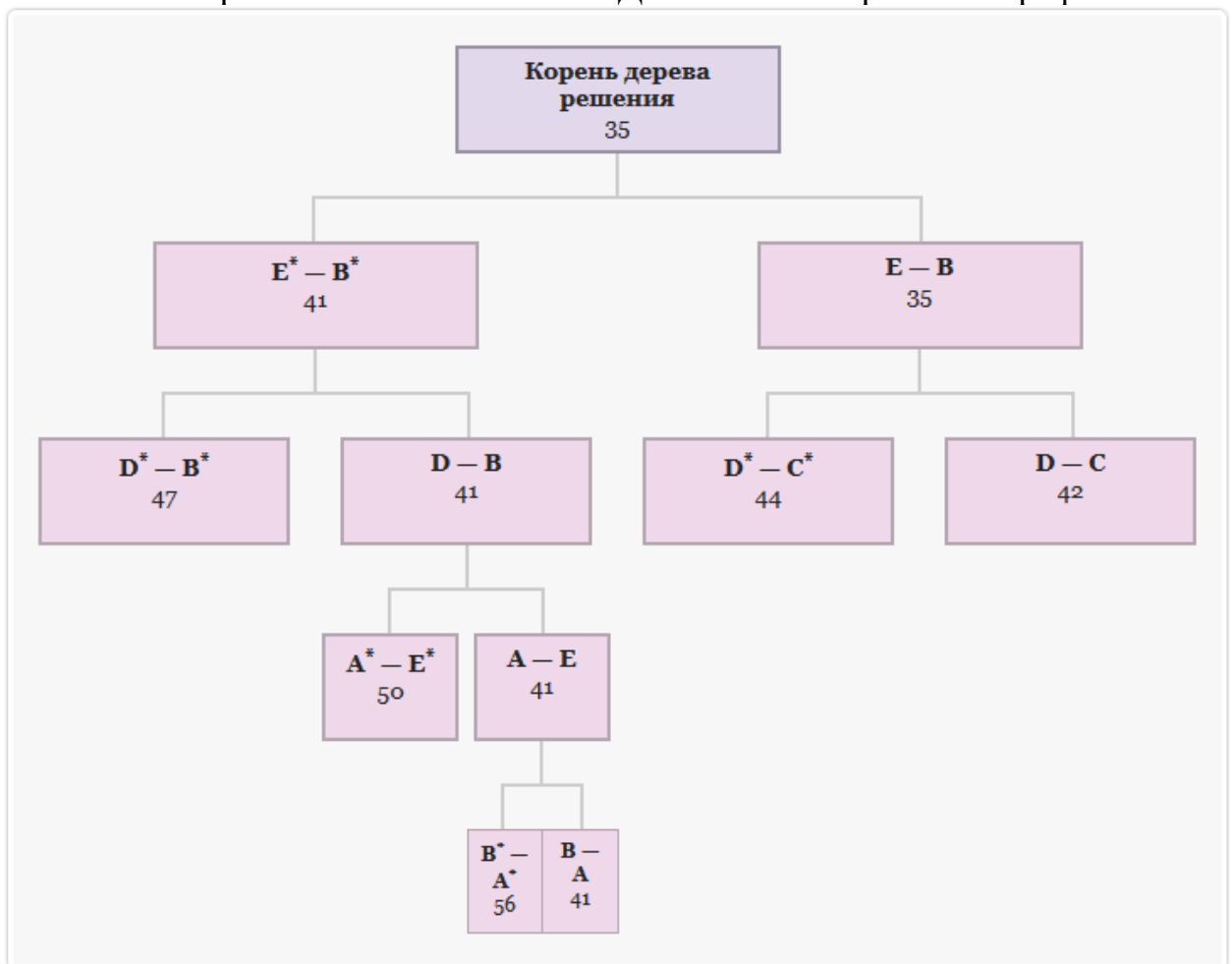
Город	С	Д
С	М	0
Е	0	0

Найдем для новой таблицы минимумы по строкам (езде 0) и столбцам (также 0). Данные, в результате редукции строк и столбцов, не изменятся.

Вычислим локальную нижнюю границу для ветви, включающей отрезок В-А:

$$H = 41 + (0 + 0) + (0 + 0) = 41.$$

В то же время локальная нижняя граница для ветви не включающей отрезок В-А равна: $H^* = 41 + 15 = 56$. Добавим оба варианта в граф.



Ветвь, включающая отрезок В-А в итоговый маршрут, имеет локальную нижнюю границу 41. Это меньше, чем соответствующие значения других ветвей, следовательно, продолжаем решение ветви с вершиной В-А.

Вычислим оценки для нулевых клеток и найдем среди них максимальную.

Город	С	Д
С	М	0 (∞)
Е	0 (∞)	0 (0)

Здесь две клетки (С-Д и Е-С) имеют максимальное значение оценки, которое бесконечно велико (напомню, что М означает бесконечно большое число, и, к примеру, в строке С оно является минимумом, т. к. саму нулевую клетку при подсчете оценки не учитываем, а других клеток в этой строке нет). Выбираем любую из этих двух ячеек, допустим, С-Д.

Проводим редукцию матрицы. Обратного пути в таблице нет, поэтому исключать его нужды нет.

Город	С
Е	0

На этом основная часть решения задачи может считаться завершенной, и мы переходим к последнему пункту 14.

14. Построение полного маршрута и определение его длины

Найдя все отрезки пути, остается только соединить их между собой и рассчитать общую длину пути (стоимость поездки по этому маршруту, затраченное время и т. д.). Расстояния между городами берем из самой первой таблицы с исходными данными.

В нашем примере в ходе решения задачи коммивояжера мы нашли следующие участки пути: D-B (17), A-E (8), B-A (5), C-D (6) и E-C (5). В скобках указано расстояние между городами. Упорядочим и соединим эти отрезки, тогда маршрут получится следующий: $A \rightarrow E \rightarrow C \rightarrow D \rightarrow B \rightarrow A$.

Общая длина пути:

$$L = \sum C_{ij} = 8 + 5 + 6 + 17 + 5 = 41.$$

Обратите внимание, что длину маршрута мы уже знали, это самое последнее значение локальной нижней границы правильной ветви решения.

Источник: *Галяутдинов Р.Р.* Задача коммивояжера - метод ветвей и границ // Сайт преподавателя экономики. [2020]. URL: <http://galyautdinov.ru/post/zadacha-kommivoyazhera> (дата обращения: 22.02.2021).

ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ

Задача 1.

Задание. Решить задачу коммивояжера.

Для решения можно использовать программу Excel.

Варианты задания 1.

1.

	A	B	C	D	E
A	М	11	12	6	18
B	5	М	5	5	5
C	18	8	М	12	20
D	11	12	11	М	17
E	14	11	5	7	М

2.

	A	B	C	D	E
A	M	14	20	14	15
B	9	M	21	15	14
C	10	17	M	8	14
D	15	21	23	M	11
E	8	7	8	8	M

3.

	A	B	C	D	E
A	M	6	5	6	5
B	13	M	12	18	12
C	12	7	M	19	13
D	12	8	15	M	6
E	9	13	19	21	M

4.

	A	B	C	D	E
A	M	27	25	19	17
B	12	M	21	14	18
C	19	25	M	13	18
D	18	24	18	M	19
E	12	12	12	12	M

5.

	A	B	C	D	E
A	M	5	5	5	5
B	12	M	11	17	11
C	11	6	M	18	12
D	11	7	14	M	5
E	8	12	18	20	M

6.

	A	B	C	D	E
--	---	---	---	---	---

A	M	4	13	10	6
B	19	M	17	7	11
C	17	11	M	10	5
D	4	5	4	M	5
E	16	10	10	11	M

7.

	A	B	C	D	E
A	M	17	15	9	5
B	2	M	11	4	8
C	9	15	M	3	8
D	8	14	8	M	9
E	2	3	2	2	M

8.

	A	B	C	D	E
A	M	5	14	11	7
B	20	M	18	8	12
C	18	12	M	11	6
D	5	5	5	M	5
E	17	11	11	12	M

9.

	A	B	C	D	E
A	M	11	12	6	18
B	5	M	5	5	5
C	18	8	M	12	20
D	11	12	11	M	17
E	14	11	5	7	M

10.

	A	B	C	D	E
A	M	8	14	8	9
B	3	M	15	9	8
C	4	11	M	2	811
D	9	15	17	M	5
E	4	4	5	4	M

11.

	A	B	C	D	E
A	M	11	17	11	12
B	6	M	18	12	11
C	7	14	M	5	11
D	12	18	20	M	8
E	5	5	5	5	M

12.

	A	B	C	D	E
A	M	17	18	14	24
B	11	M	11	11	12
C	24	14	M	18	26
D	17	18	17	M	23
E	20	17	11	13	M

13.

	A	B	C	D	E
A	M	13	19	13	14
B	8	M	20	14	13
C	9	16	M	7	13
D	14	20	22	M	10
E	7	6	7	7	M

14.

	A	B	C	D	E
A	M	9	9	9	10
B	16	M	15	21	15
C	15	11	M	23	17
D	16	12	19	M	10
E	13	17	23	25	M

15.

	A	B	C	D	E
A	M	25	23	17	13
B	10	M	19	12	16
C	17	23	M	11	16
D	16	22	16	M	17
E	9	10	10	10	M

16.

	A	B	C	D	E
A	M	7	7	7	7
B	14	M	13	19	13
C	13	9	M	21	15
D	14	10	17	M	8
E	11	15	21	21	M

17.

	A	B	C	D	E
A	M	22	20	14	10
B	7	M	16	9	13
C	14	20	M	8	13
D	13	19	13	M	14
E	7	6	7	7	M

18.

	A	B	C	D	E
A	M	8	17	14	10
B	23	M	21	11	15
C	21	15	M	14	9
D	8	8	8	M	8
E	20	14	14	15	M

19.

	A	B	C	D	E
A	M	18	16	10	22
B	9	M	9	9	9
C	18	12	M	16	24
D	15	16	15	M	21
E	18	15	9	11	M

20.

	A	B	C	D	E
A	M	10	19	16	12
B	25	M	23	13	17
C	23	17	M	16	11
D	10	10	10	M	10
E	22	16	16	17	M

21.

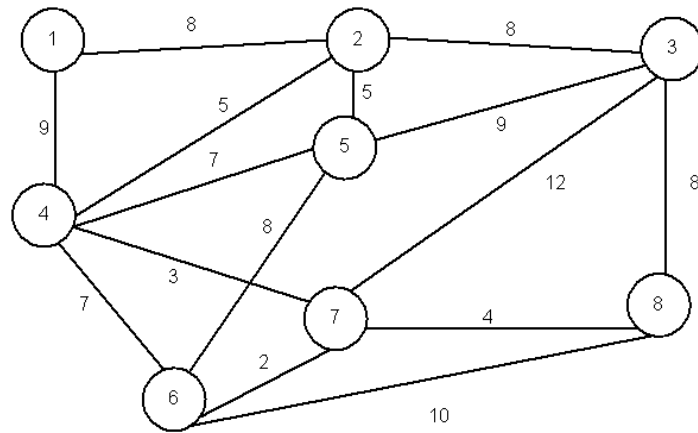
	A	B	C	D	E
A	M	12	20	1113	16
B	6	M	22	15	16
C	11	13	M	10	10
D	1714	20	21	M	11
E	10	9	10	10	M

Задание 2

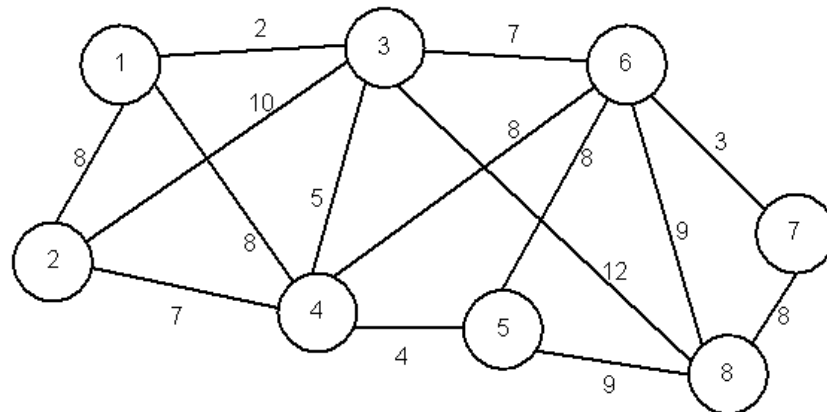
Задание. С помощью алгоритма Дейкстры найти кратчайший маршрут из вершины 1 до остальных вершин, и длины кратчайших маршрутов. Указать порядок присоединения вершин и выделить те вершины, для которых в процессе осуществления алгоритма Дейкстры менялись значения.

Варианты задания 2.

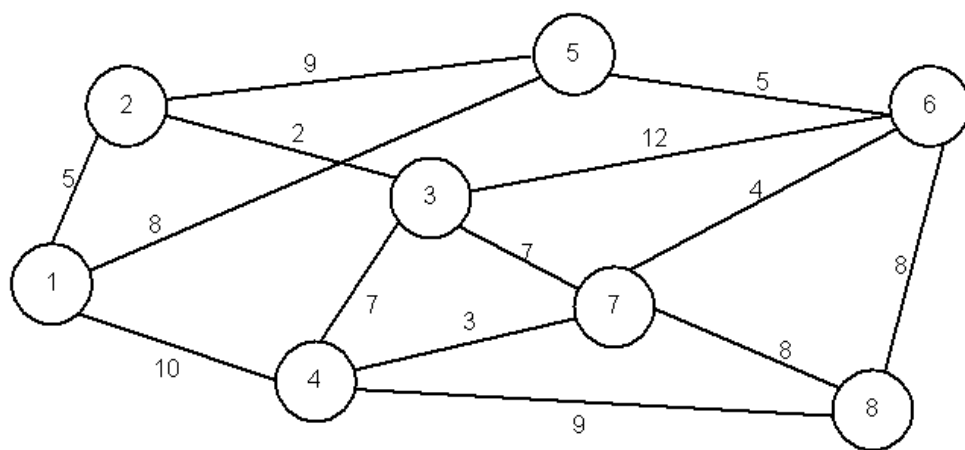
1.



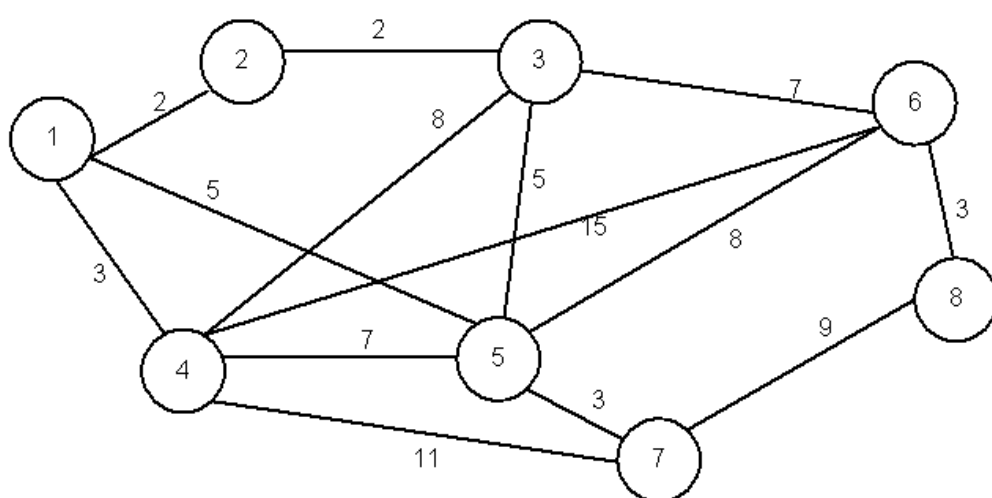
2.



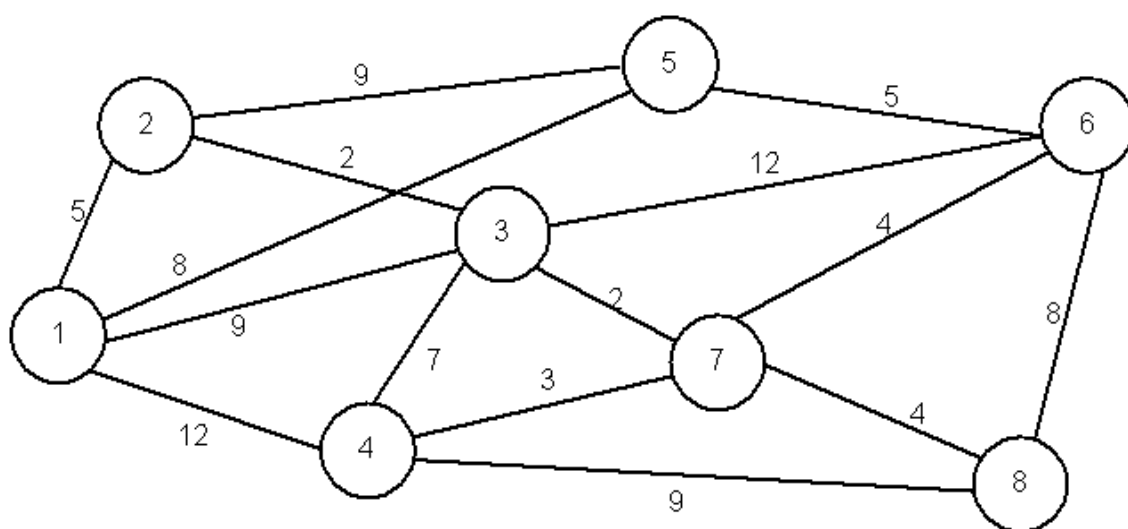
3.



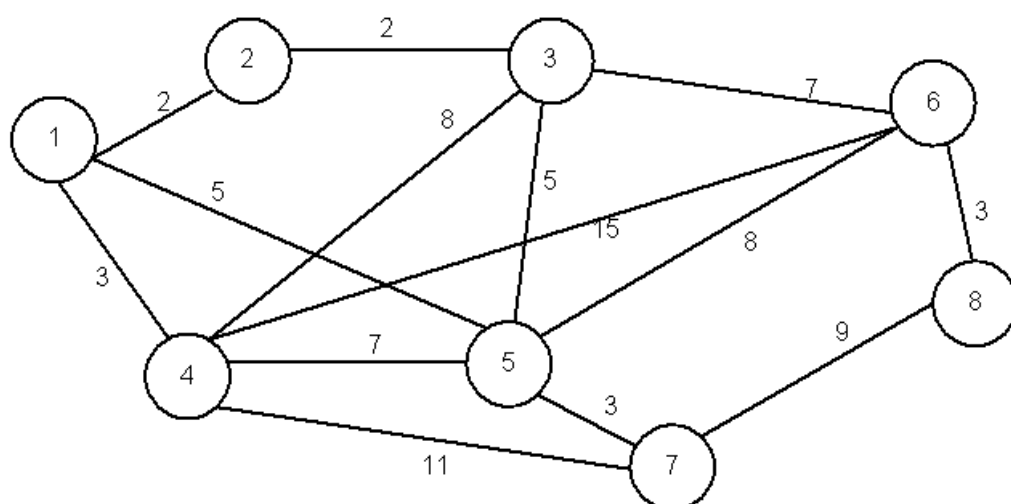
4.



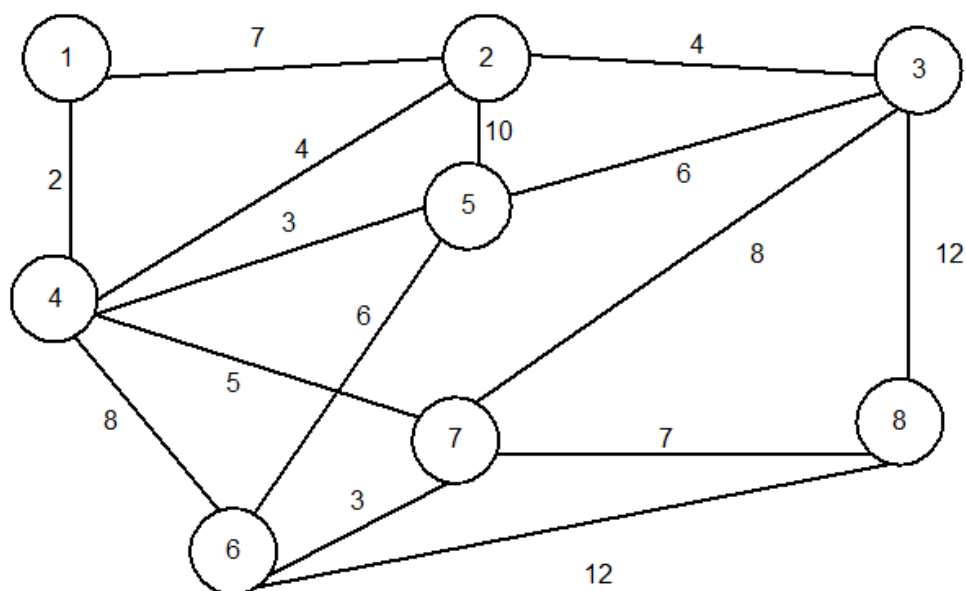
5.



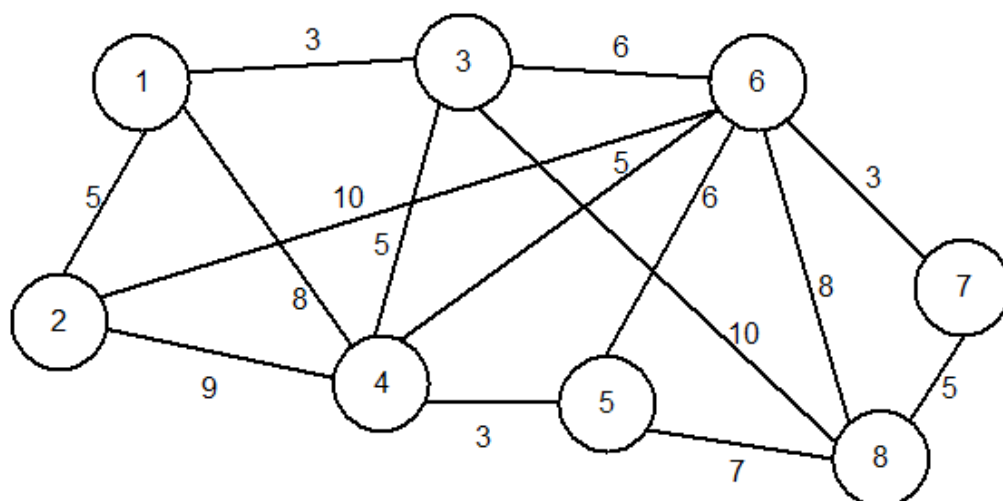
6.



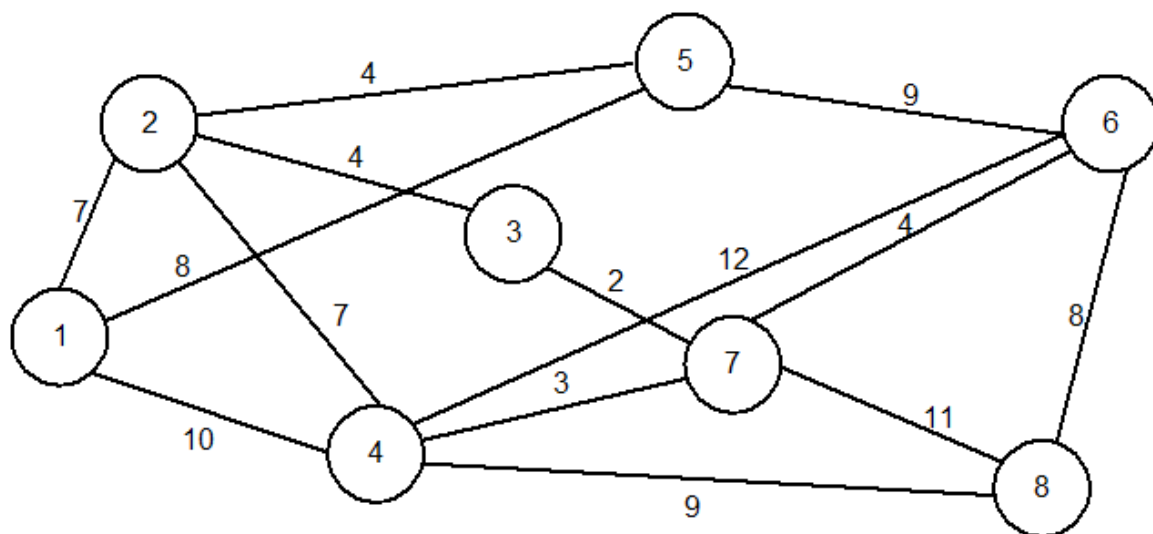
7.



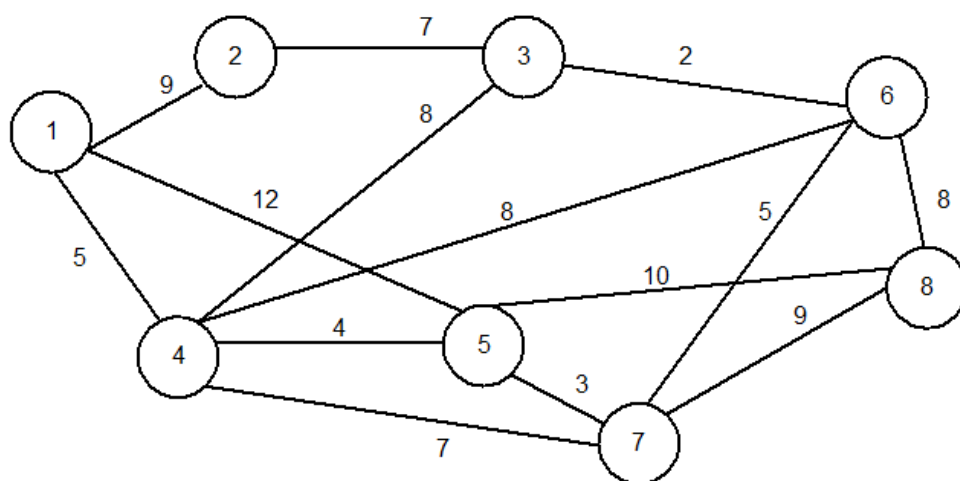
8.



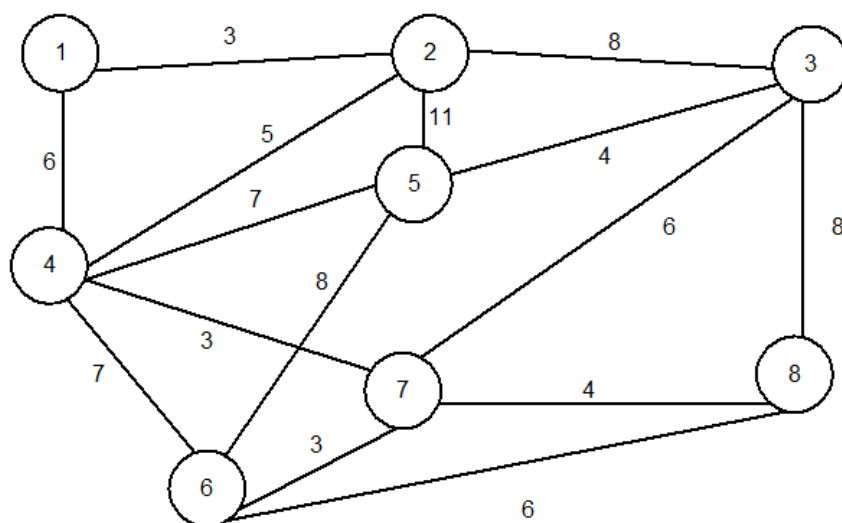
9.



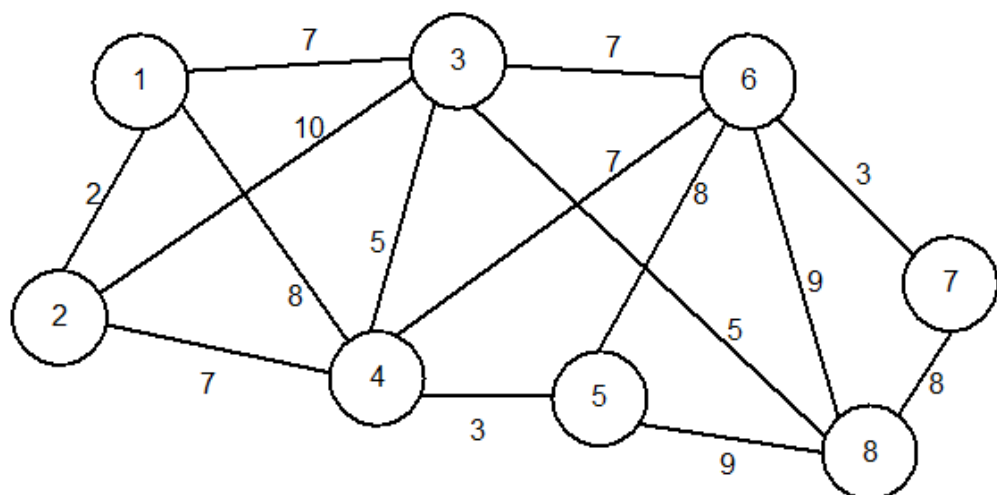
10.



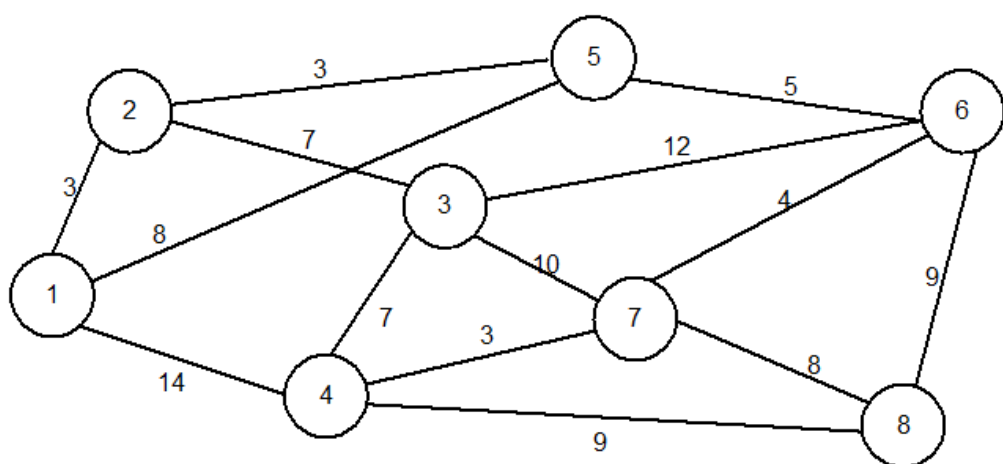
11.



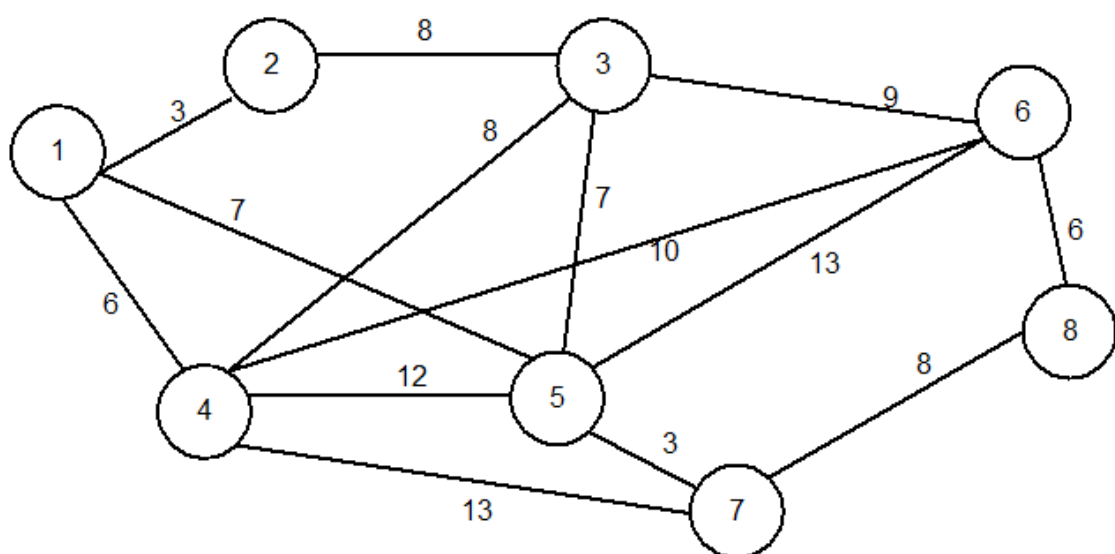
12.



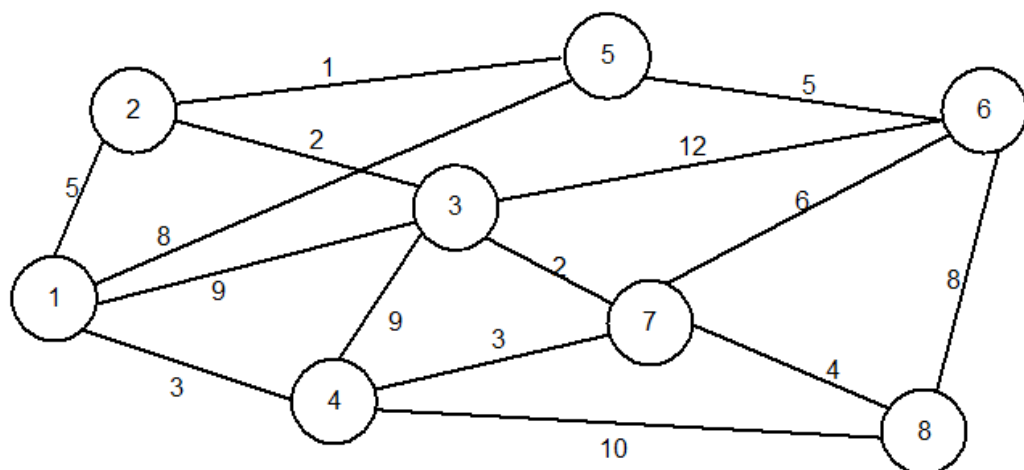
13.



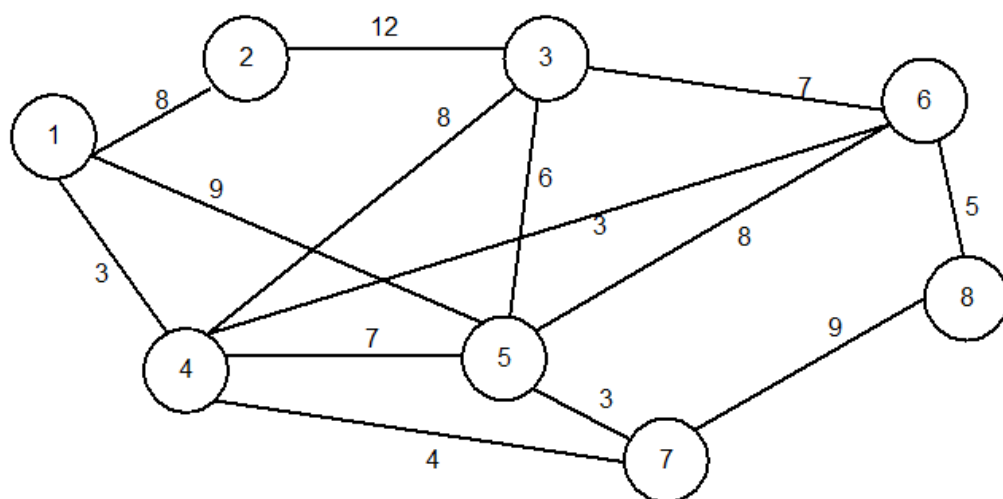
14.



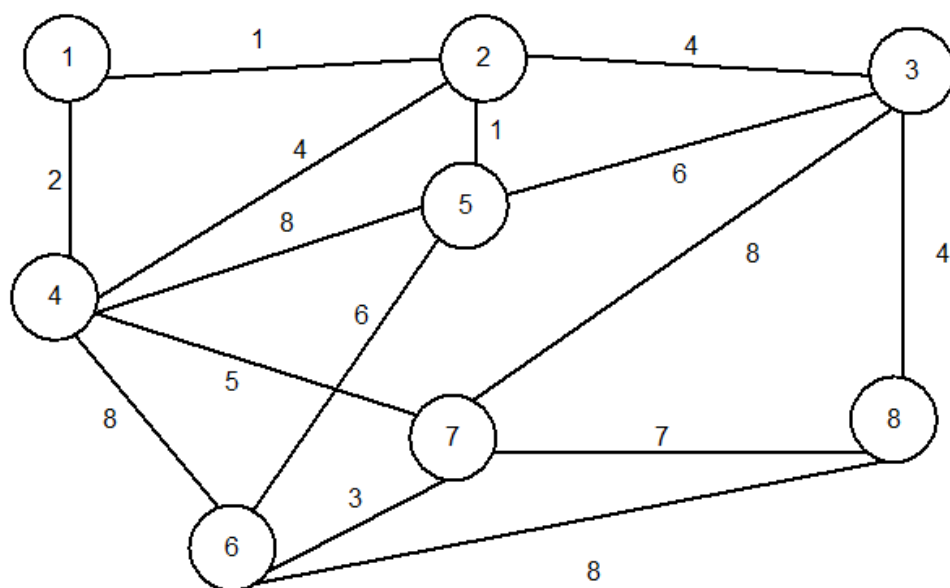
15.



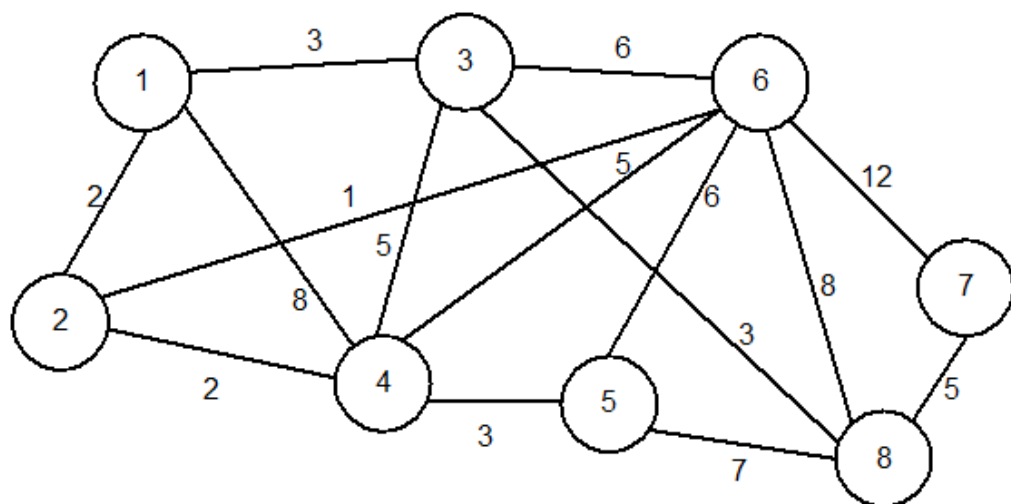
16.



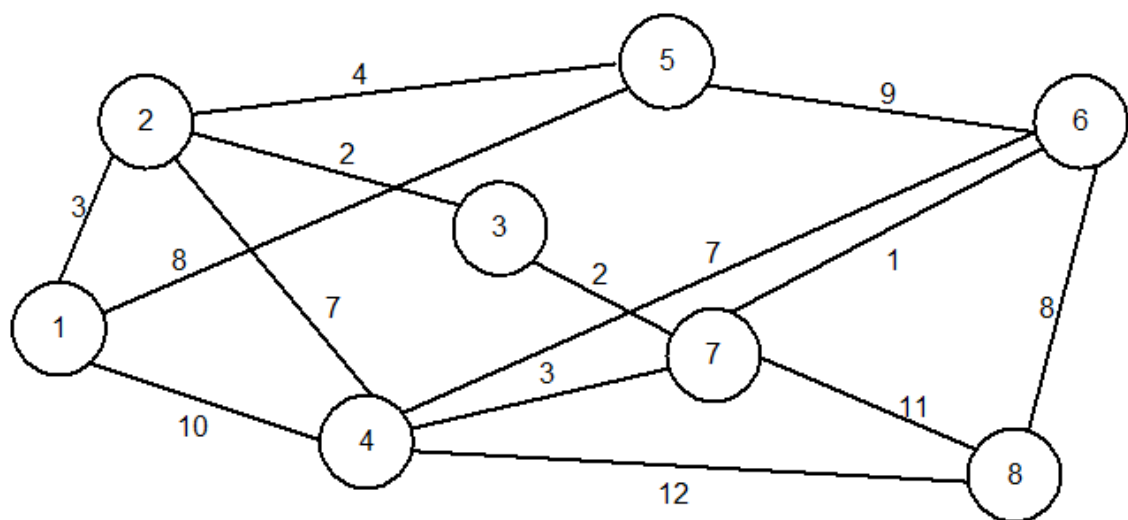
17.



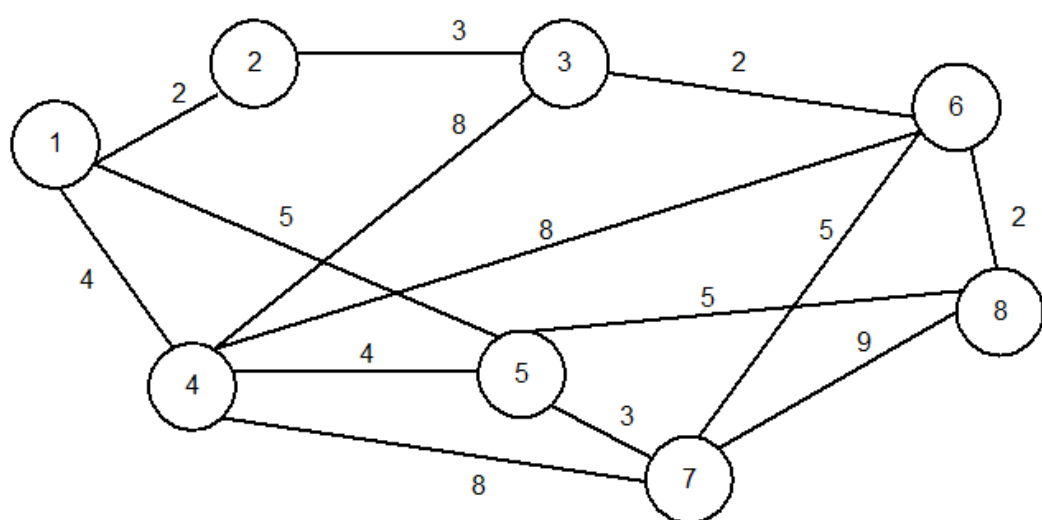
18.



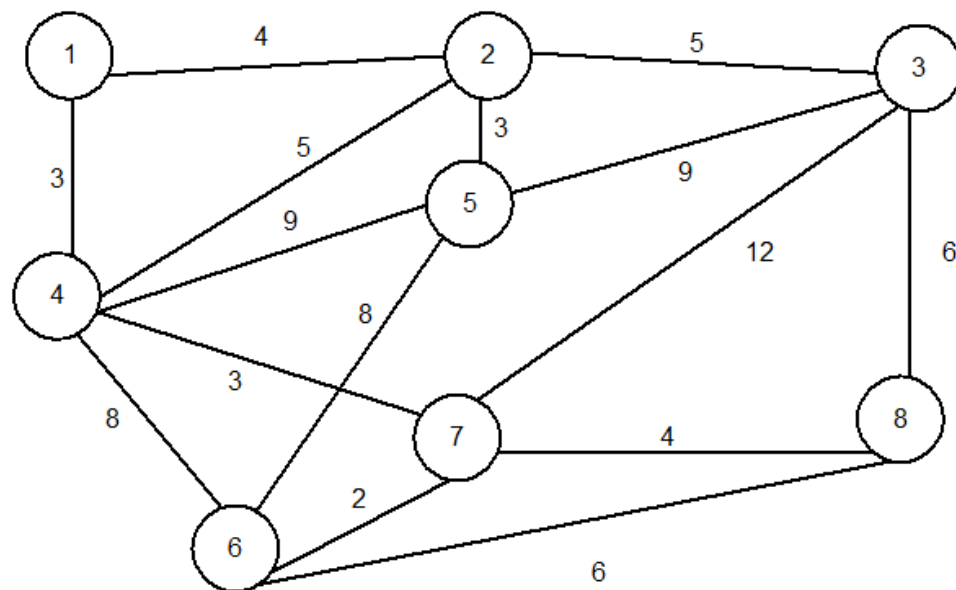
19.



20.



21.



Задание 3

Задание. С помощью алгоритма Прима найти минимальное остовное дерево для графа, представленного матрицей весов, нарисовать исходный граф и выделить полученное минимальное остовное (покрывающее) дерево.

Варианты задания 3.

1.	2.	3.
$\begin{pmatrix} - & 8 & 4 & 6 & 9 & \infty \\ 8 & - & 5 & \infty & 6 & \infty \\ 4 & 5 & - & 4 & 3 & \infty \\ 6 & \infty & 4 & - & 5 & 12 \\ 9 & 6 & 3 & 5 & - & 10 \\ \infty & \infty & \infty & 12 & 10 & - \end{pmatrix}$	$\begin{pmatrix} - & 7 & 5 & 3 & 8 & 4 \\ 7 & - & 6 & \infty & 4 & \infty \\ 5 & 6 & - & 5 & 8 & \infty \\ 3 & \infty & 5 & - & 5 & 9 \\ 8 & 4 & 8 & 5 & - & 10 \\ 4 & \infty & \infty & 9 & 10 & - \end{pmatrix}$	$\begin{pmatrix} - & 10 & 3 & 6 & 9 & \infty \\ 10 & - & 5 & \infty & 8 & 7 \\ 3 & 5 & - & 5 & 8 & 3 \\ 6 & \infty & 5 & - & 4 & 9 \\ 9 & 8 & 8 & 4 & - & 7 \\ \infty & 7 & 3 & 9 & 7 & - \end{pmatrix}$

4. $\begin{pmatrix} - & 2 & 5 & \infty & 8 & 4 \\ 2 & - & 6 & 3 & 10 & \infty \\ 5 & 6 & - & 5 & 8 & \infty \\ \infty & 3 & 5 & - & 5 & 9 \\ 8 & 10 & 8 & 5 & - & 5 \\ 4 & \infty & \infty & 9 & 5 & - \end{pmatrix}$	5. $\begin{pmatrix} - & 9 & 7 & 6 & 8 & \infty \\ 9 & - & 5 & \infty & 6 & \infty \\ 7 & 5 & - & 2 & 3 & \infty \\ 6 & \infty & 2 & - & 4 & 11 \\ 8 & 6 & 3 & 4 & - & 10 \\ \infty & \infty & \infty & 11 & 10 & - \end{pmatrix}$	6. $\begin{pmatrix} - & \infty & 8 & 3 & 8 & 4 \\ \infty & - & 6 & 7 & 4 & \infty \\ 8 & 6 & - & 5 & 8 & \infty \\ 3 & 7 & 5 & - & 3 & 9 \\ 8 & 4 & 8 & 3 & - & 10 \\ 4 & \infty & \infty & 9 & 10 & - \end{pmatrix}$
7. $\begin{pmatrix} - & 9 & 3 & 10 & 9 & \infty \\ 9 & - & 5 & \infty & 8 & 11 \\ 3 & 5 & - & 5 & \infty & 3 \\ 10 & \infty & 5 & - & 4 & 9 \\ 9 & 8 & \infty & 4 & - & 12 \\ \infty & 11 & 3 & 9 & 12 & - \end{pmatrix}$	8. $\begin{pmatrix} - & 6 & 5 & \infty & 11 & 4 \\ 6 & - & 6 & 3 & 2 & \infty \\ 5 & 6 & - & 5 & 8 & \infty \\ \infty & 3 & 5 & - & 5 & 3 \\ 11 & 2 & 8 & 5 & - & 5 \\ 4 & \infty & \infty & 3 & 5 & - \end{pmatrix}$	9. $\begin{pmatrix} - & 12 & 5 & 3 & 8 & 9 \\ 12 & - & 8 & \infty & 4 & \infty \\ 5 & 8 & - & 5 & 8 & \infty \\ 3 & \infty & 5 & - & 5 & 9 \\ 8 & 4 & 8 & 5 & - & 2 \\ 9 & \infty & \infty & 9 & 2 & - \end{pmatrix}$
10. $\begin{pmatrix} - & 3 & 10 & 6 & 2 & \infty \\ 3 & - & 5 & \infty & 8 & 6 \\ 10 & 5 & - & 5 & 8 & 3 \\ 6 & \infty & 5 & - & 10 & 9 \\ 2 & 8 & 8 & 10 & - & 7 \\ \infty & 6 & 3 & 9 & 7 & - \end{pmatrix}$	11. $\begin{pmatrix} - & 3 & 4 & 6 & 13 & \infty \\ 3 & - & 5 & \infty & 6 & \infty \\ 4 & 5 & - & 8 & 3 & \infty \\ 6 & \infty & 8 & - & 5 & 9 \\ 13 & 6 & 3 & 5 & - & 10 \\ \infty & \infty & \infty & 9 & 12 & 10 & - \end{pmatrix}$	12. $\begin{pmatrix} - & 4 & 6 & 3 & 8 & 4 \\ 4 & - & 6 & \infty & 4 & \infty \\ 6 & 6 & - & 2 & 8 & \infty \\ 3 & \infty & 2 & - & 5 & 9 \\ 8 & 4 & 8 & 5 & - & 8 \\ 4 & \infty & \infty & 9 & 8 & - \end{pmatrix}$
13. $\begin{pmatrix} - & 2 & 3 & 6 & 3 & \infty \\ 2 & - & 5 & \infty & 8 & 7 \\ 3 & 5 & - & 5 & 8 & 3 \\ 6 & \infty & 5 & - & 4 & 9 \\ 3 & 8 & 8 & 4 & - & 8 \\ \infty & 7 & 3 & 9 & 8 & - \end{pmatrix}$	14. $\begin{pmatrix} - & 9 & 5 & \infty & 8 & 4 \\ 9 & - & 6 & 3 & 10 & \infty \\ 5 & 6 & - & 5 & 8 & \infty \\ \infty & 3 & 5 & - & 15 & 9 \\ 8 & 10 & 8 & 15 & - & 3 \\ 4 & \infty & \infty & 9 & 3 & - \end{pmatrix}$	15. $\begin{pmatrix} - & 4 & 7 & 6 & 8 & \infty \\ 4 & - & 5 & \infty & 6 & \infty \\ 7 & 5 & - & 2 & 3 & \infty \\ 6 & \infty & 2 & - & 4 & 2 \\ 8 & 6 & 3 & 4 & - & 10 \\ \infty & \infty & \infty & 2 & 10 & - \end{pmatrix}$

16. $\begin{pmatrix} - & \infty & 7 & 3 & 8 & 4 \\ \infty & - & 6 & 7 & 4 & \infty \\ 7 & 6 & - & 4 & 8 & \infty \\ 3 & 7 & 4 & - & 3 & 9 \\ 8 & 4 & 8 & 3 & - & 2 \\ 4 & \infty & \infty & 9 & 2 & - \end{pmatrix}$	17. $\begin{pmatrix} - & 2 & 3 & 5 & 9 & \infty \\ 2 & - & 5 & \infty & 8 & 11 \\ 3 & 5 & - & 5 & \infty & 3 \\ 5 & \infty & 5 & - & 4 & 9 \\ 9 & 8 & \infty & 4 & - & 7 \\ \infty & 11 & 3 & 9 & 7 & - \end{pmatrix}$	18. $\begin{pmatrix} - & 3 & 5 & \infty & 11 & 4 \\ 3 & - & 6 & 3 & 2 & \infty \\ 5 & 6 & - & 5 & 7 & \infty \\ \infty & 3 & 5 & - & 5 & 3 \\ 11 & 2 & 7 & 5 & - & 5 \\ 4 & \infty & \infty & 3 & 5 & - \end{pmatrix}$
19. $\begin{pmatrix} - & 2 & 5 & 3 & 8 & 9 \\ 2 & - & 8 & \infty & 14 & \infty \\ 5 & 8 & - & 15 & 8 & \infty \\ 3 & \infty & 15 & - & 5 & 9 \\ 8 & 14 & 8 & 5 & - & 12 \\ 9 & \infty & \infty & 9 & 12 & - \end{pmatrix}$	20. $\begin{pmatrix} - & 13 & 10 & 16 & 2 & \infty \\ 13 & - & 5 & \infty & 8 & 6 \\ 10 & 5 & - & 5 & 8 & 3 \\ 16 & \infty & 5 & - & 11 & 9 \\ 2 & 8 & 8 & 11 & - & 17 \\ \infty & 6 & 3 & 9 & 17 & - \end{pmatrix}$	21. $\begin{pmatrix} - & 18 & 4 & 16 & 9 & \infty \\ 18 & - & 5 & \infty & 6 & \infty \\ 4 & 5 & - & 14 & 13 & \infty \\ 16 & \infty & 14 & - & 5 & 12 \\ 9 & 6 & 13 & 5 & - & 10 \\ \infty & \infty & \infty & 12 & 10 & - \end{pmatrix}$