МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования
«Юго-Западный государственный университет»
(ЮЗГУ)

Кафедра программной инженерии

УТВЕРЖДАЮ:
Проректор по учебной работе
________ О.Г. Локтионова
« 28 » 10 2020 г.

ПОСТРОЕНИЕ МИНИМАЛЬНОГО ОСТОВНОГО ДЕРЕВА

Методические указания для выполнения лабораторной работы по дисциплине «Дискретная математика» для студентов направления подготовки 09.03.04 Программная инженерия

Курск - 2020

УДК 519.17

Составитель: Р.А. Томакова

Рецензент
Кандидат технических наук, доцент *А.В.Малышев*

Построение минимального остовного дерева: методические указания для выполнения лабораторной работы по дисциплине «Дискретная математика» для студентов направления подготовки 09.03.04 Программная инженерия/ Юго-Зап. гос. ун-т; сост. Р.А. Томакова. Курск, 2020. 17 с.

Составлены в соответствии с федеральным государственным образовательным стандартом высшего образования направления подготовки 09.03.04 Программная инженерия и на основании учебного плана направления подготовки 09.03.04 Программная инженерия.

В методических указаниях представлены основные понятия и свойства графов, необходимые для построения минимального остовного дерева. Сформулированы требования для выполнения лабораторной работы по дисциплине «Дискретная математика», разобраны примеры выполнения заданий, приведены вопросы к защите.

Предназначены для студентов, обучающихся направления подготовки 09.03.04 Программная инженерия (профиль «Разработка программно-информационных систем») всех форм обучения.

Текст печатается в авторской редакции

Подписано в печать 28.10.20 . Формат 60×84 1/16.
Усл. печ. л. 2,2 . Уч.- изд. л. 2,0. Тираж 25 экз. Заказ. 1390. Бесплатно.
Юго-Западный государственный университет.
305040, г. Курск, ул. 50 лет Октября, 94.

ЛАБОРАТОРНАЯ РАБОТА

ПОСТРОЕНИЕ МИНИМАЛЬНОГО ОСТОВНОГО ДЕРЕВА

Цель работы:

1. Изучение основных понятий теории графов, предназначенных для реализаций профессиональной деятельности;
2. Приобретение навыков построения оптимальных структур на основе графовых моделей;
3. Приобретение навыков разработки алгоритма Краскала для построения остовного дерева минимальной длины.
4. Научиться находить доминирующие и независимые множества.

ЗАДАНИЕ:

1. Составить матрицу пропускных возможностей графа размерностью 7х7, положив пропускные возможности $a_{12}=n$, где n-порядковый номер студента в списке журнала группы. Значения остальных a_{ij} выбрать произвольно.
2. По составленной матрице восстановить граф.
3. Для данного графа построить произвольное остовное дерево и соответствующее ему ко-дерево.
4. Реализовать алгоритм Краскала, построив остовное дерево минимальной длины.
5. Определить все доминирующие множества исходного графа, найти число доминирования.
6. Представить данный граф как неориентированный. Определить в нем независимые множества, число независимости.

ОСНОВНЫЕ ТЕОРЕТИЧЕСКИЕ ПОЛОЖЕНИЯ

Одним из наиболее важных понятий теории графов, которое часто используется в прикладных задачах, является дерево. Понятие дерева как математического объекта впервые было предложено Кирхгофом при анализе электрических цепей.

Пусть $G(X, A)$ есть неориентированный граф. Будем говорить, что $G(X, A)$ есть ***неориентированное дерево***, если справедливо любое из условий:

1) это связный граф, не имеющий циклов;

2) это связный граф, содержащий n вершин и n-1 ребер;

3) это граф, в котором каждая пара вершин соединена одной и только одной простой цепью.

Легко показать, что эти три определения эквивалентны друг другу.

Остовным деревом для неориентированного графа $G(X, A)$ с n вершинами называется каждый остовный подграф графа G, являющийся деревом (в смысле приведенного выше определения).

Пример. Рассмотрим неориентированный граф $G(X, A)$, представленный на рисунке 1.

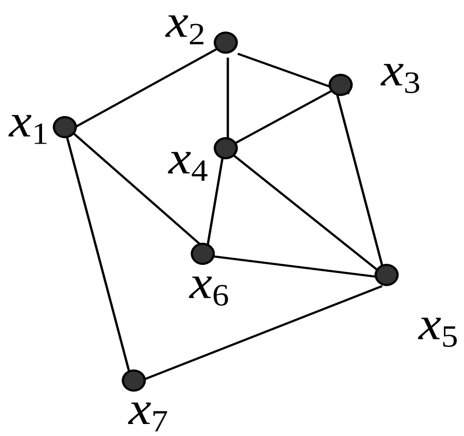


Рисунок 1- Неориентированный граф $G(X, A)$

Остовными деревьями для неориентированного графа $G(X, A)$ будут, например, остовные подграфы, приведенные на рисунке 2.

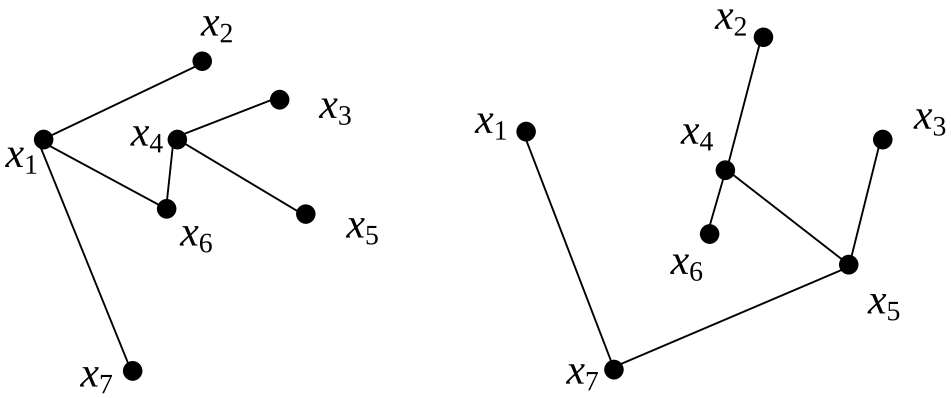


Рисунок 2. Остовные деревья для графа $G(X, A)$

Ориентированное дерево представляет собой ориентированный граф без циклов, в котором полустепень захода каждой вершины, за исключением одной (например, вершины x_1), равна единице, полустепень захода вершины x_1 (называемой корнем этого дерева) равна нулю.

Из этого определения следует, что ориентированное дерево с n вершинами имеет n-1 дуг и связно. Очевидно, что не всякий ориентированный граф содержит остовное ориентированное дерево. Например, граф, изображенный на рисунке 3, не имеет ориентированного остовного дерева.

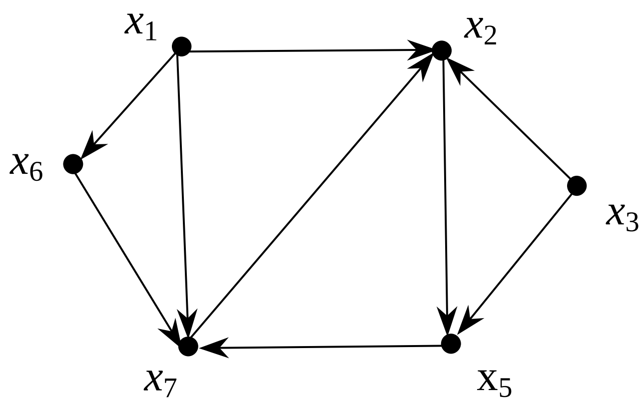


Рисунок 3. Граф, не содержащий ориентированного остовного дерева

Следует отметить, что любое неориентированное дерево можно преобразовать в ориентированное. Для этого нужно взять его произвольную вершину в качестве корня и ребрам приписать такую ориентацию, чтобы каждая вершина соединялась с корнем только одной простой цепью. На рисунке 4 показано ориентированное дерево.

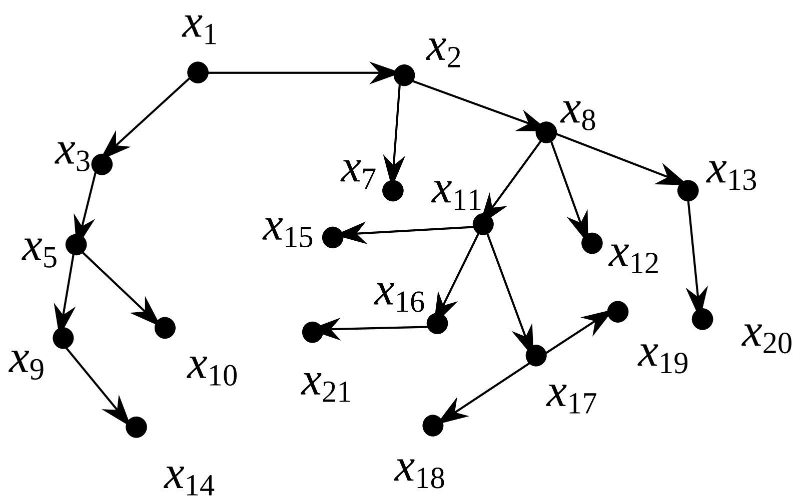


Рисунок 4- Ориентированное дерево

1. Кодерево T для графа G(X, A)*

Пусть дан неориентированный граф $G(X, A)$ и T есть остовное древо этого графа (рис.5).

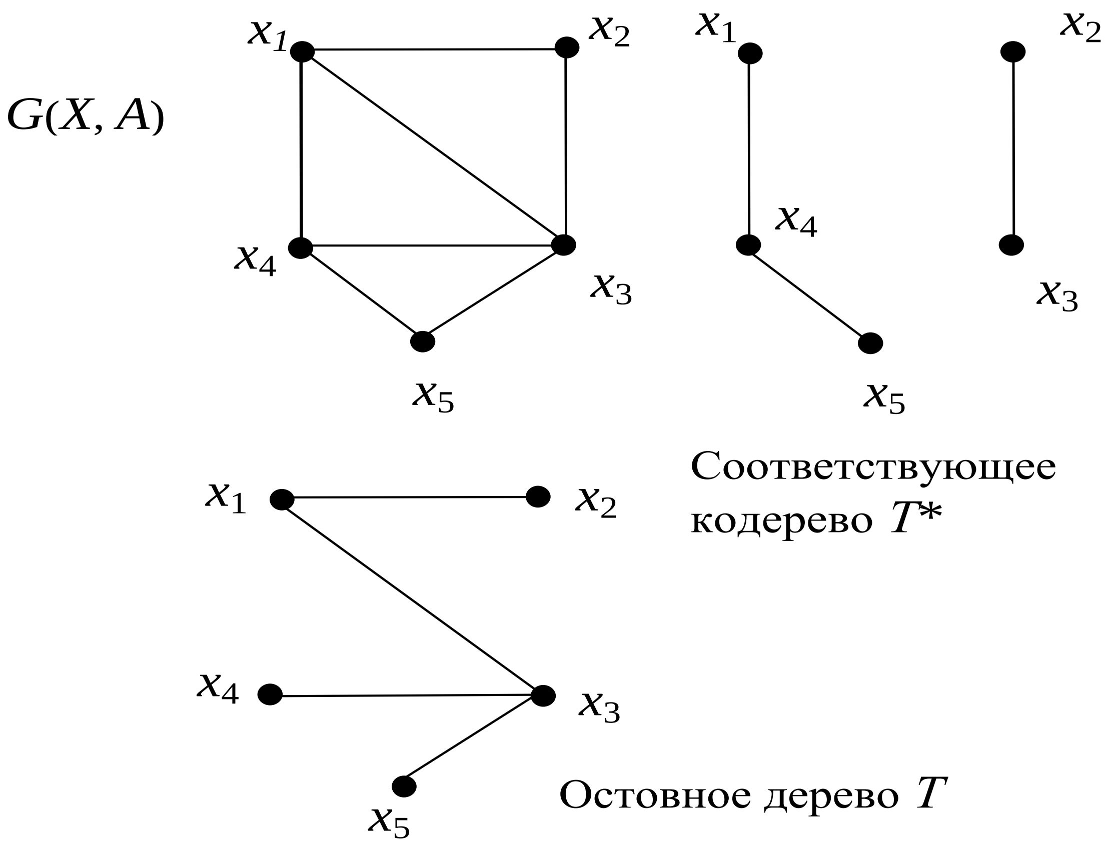


Рисунок 5. Граф $G(X, A)$, остовное дерево T и соответствующее ему кодерево T^*

Тогда кодеревом T^* остова T является подграф графа $G(X, A)$, содержащий все вершины G и только те ребра, которые не входят в T.

Кодерево может быть несвязным. Любое остовное дерево однозначно определяет свое кодерево T^*. Определим для графа $G(X, A)$ остовное дерево и соответствующее ему кодерево (рис. 5).

«Генеалогическое дерево», в котором вершины соответствуют лицам мужского пола, а дуги ориентированы от родителей к детям, представляет собой известный пример ориентированного дерева. Корень в этом дереве соответствует «основателю» рода.

В некоторых случаях возникает необходимость в построении полного списка остовных деревьев графа G. Например, если нужно отобрать «наилучшее» дерево, а критерий, позволяющий осуществить такой отбор, является очень сложным или субъективным.

Число различных остовов полного связного неориентированного помеченного графа с n вершинами равно n^{n-2}. Если $G(X, A)$ – n-вершинный граф без петель и B_0 – его матрица инциденций, с одной удаленной строкой (то есть с n-1 независимыми строками), а B_0^T – транспонированная матрица к B_0, то определитель $det\ |B_0 \cdot B_0^T|$ равен числу различных остовных деревьев графа $G(X, A)$.

2. *Деревья с минимальной длиной взвешенных путей*

m-деревом назовем ориентированное дерево, в котором полустепень исхода каждой вершины не превышает m. Заметим, что в ориентированном дереве полустепень захода для корня равна нулю, а для остальных вершин – единице. Обычно m-дерево изображают так, что корень находится сверху, а все ребра направлены вниз. Все вершины, находящиеся на одном и том же расстоянии от корня, лежат на одной горизонтальной линии.

На рисунке 6 изображено 3-дерево.

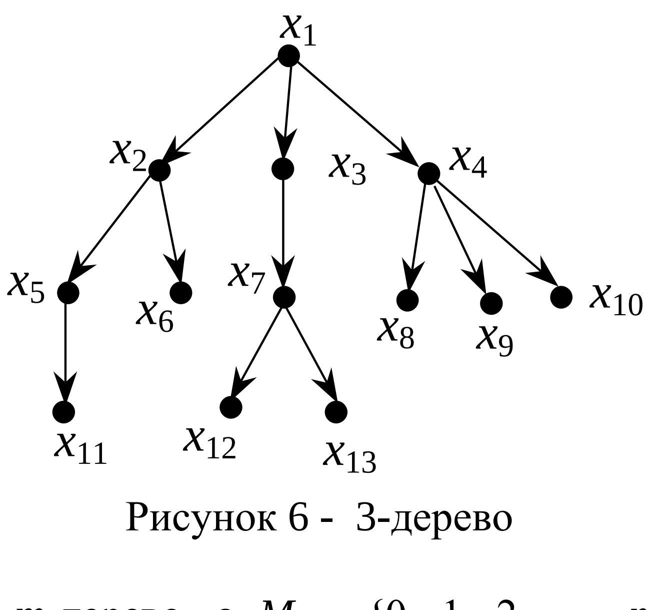


Рисунок 6 - 3-дерево

Пусть T – m-дерево, а M = {0, 1, 2, …, m-1} – алфавит, состоящий из m букв. Предположим, что мы приписываем каждому ребру в дереве T букву из алфавита так, что не может быть двух ребер, исходящих из одной и той же вершины и помеченных одной и той же буквой. Тогда мы можем приписать каждой вершине дерева *T* слово, образуемое ***конкатенацией*** букв, которыми помечены ребра, встречающиеся при движении из корня в данную вершину. Такие слова, приписанные листьям дерева *T*, называют ***кодовыми словами***, и говорят, что они образуют ***префиксный код***.

Например, в случае тернарного дерева, изображенного на рисунке 7, кодовыми словами являются 00, 010, 120, 121, 22, 20, 21.

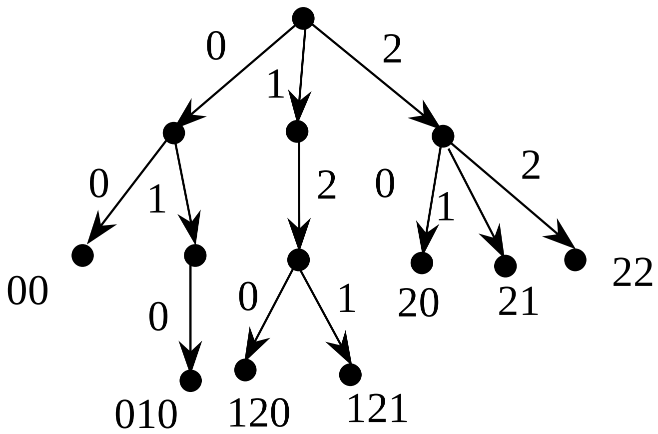


Рисунок 7- Тернарное дерево с кодовыми словами

Интересным и полезным свойством этих слов в префиксном коде является то, что никакое из слов не совпадает с началом другого слова.

Предположим, что мы закодировали L сообщений M_1, M_2, …, M_L словами префиксного кода. Если мы передадим последовательность каких-либо из этих закодированных сообщений по панелям связи, то мы получим на принимающем конце канала последовательность букв, образуемую конкатенацией кодовых слов, соответствующих переданным сообщениям. Чтобы из этой последовательности выделить сообщение, необходимо разложить на слова из префиксного кода.

Процесс разложения последовательности на кодовые слова называется ***декодированием***. И его можно осуществить с помощью дерева, соответствующего префиксному коду. Рассмотрим последовательность 120202200, которая образована конкатенацией некоторых слов префиксного кода, соответствующего дереву, изображенному на рисунке 7.

Чтобы декодировать эту последовательность, рассмотрим составляющие ее буквы слева направо. В соответствии с этим просмотром мы двигаемся по дереву, начиная с корня вдоль ребер, которые соответствуют рассматриваемым буквам, до тех пор, пока мы не дойдем до листа дерева. Кодовое слово, соответствующее этому листу является первым словом в данной последовательности. Таким образом, мы получаем 120 в качестве первого слова в последовательности 120202200. Затем повторяем процесс декодирования для оставшейся последовательности 202200 и выделяем 20, 22, 00 в качестве второго, третьего и четвертого кодовых слов в последовательности 120202200.

Из процесса декодирования следует, что стоимость декодирования кодового слова пропорциональна числу букв в этом слове.

Если w_i – частота появления сообщения M_i, то ожидаемая стоимость декодирования зависит от суммы $\sum_{i=1}^{L} w_i l_i$, где l_i – длина пути от корня до листа, соответствующего сообщению M_i = $\sum_{i=1}^{L} w_i l_i$, называется ***длиной взвешенных путей*** дерева.

Таким образом, ожидаемую стоимость декодирования можно свести к минимуму таким выбором длин кодовых слов, что полученное дерево будет иметь минимальную длину взвешенных путей. Возникает задача построения m-дерева с минимальной длиной взвешенных путей на заданном множестве весов w_1, w_2, … , w_L.

3. *Кратчайший остов графа*

Рассмотрим взвешенный связный неориентированный граф $G(X, A)$: вес ребра (x_i, x_j) обозначим c_{ij}. Из большого числа остовов нужно найти один, у которого сумма весов ребер наименьшая.

Такая задача возникает, например, в случае, когда вершины являются клеммами электрической сети, которые должны быть соединены друг с другом с помощью проводов наименьшей длины (для уменьшения уровня наводок).

Алгоритм Краскала.

1. Пусть дан связный граф, содержащий n вершин.
2. Упорядочить ребра в порядке неубывания их весов.
3. Начать с первого ребра в этом списке, добавлять ребра в графе T, соблюдая условие: такое добавление не должно приводить к появлению цикла в $G(X, A)$.

4. Шаг повторять до тех пор, пока число ребер в графе не станет n-1.

4. Независимые множества

Рассмотрим неориентированный граф $G(X, \Gamma)$. Независимое множество вершин (или ***внутреннее устойчивое множество***) есть такое множество вершин графа G, что любые две вершины в нем несмежны, то есть никакая пара вершин не соединена ребром. Следовательно, любое множество $S \subset X$, которое удовлетворяет условию $S \cap \Gamma(S) = \varnothing$, является ***независимым множеством***.

Для неориентированного графа $G(X, \Gamma)$, представленного на рисунке 8, $\{x_2, x_5, x_7, x_8\}$ – независимое множество.

Независимое множество называется ***максимальным***, когда нет другого независимого множества, в которое оно входило бы.

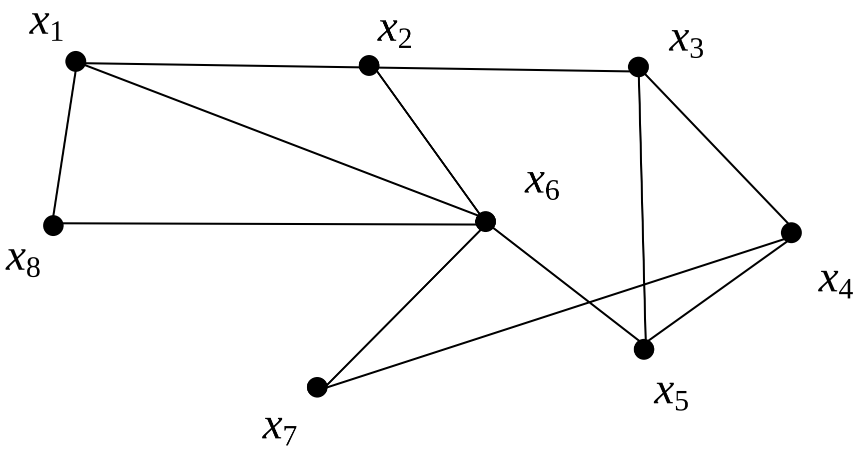


Рисунок 8 - Граф $G(X, \Gamma)$

Числом независимости графа $G(X, \Gamma)$ называется число элементов в максимальном независимом множестве данного графа. $\alpha(G) = 4$ для данного графа.

5. Доминирующее множество

Для ориентированного графа $G(X, \Gamma)$ доминирующее множество вершин (или ***внешне устойчивое множество***) есть множество вершин $S \subseteq X$, выбранное так, что для каждой вершины x_j, не входящей в S, существует дуга, идущая из некоторой вершины множества S в x_j.

Таким образом, S – ***доминирующее множество*** вершин, если $S \cup \Gamma(S) = X$.

Для графа $G(X, \Gamma)$, представленного на рисунке 9, множества вершин $\{x_1, x_4, x_6\}$, $\{x_1, x_4\}$, $\{x_3, x_5, x_6\}$ являются доминирующими множествами.

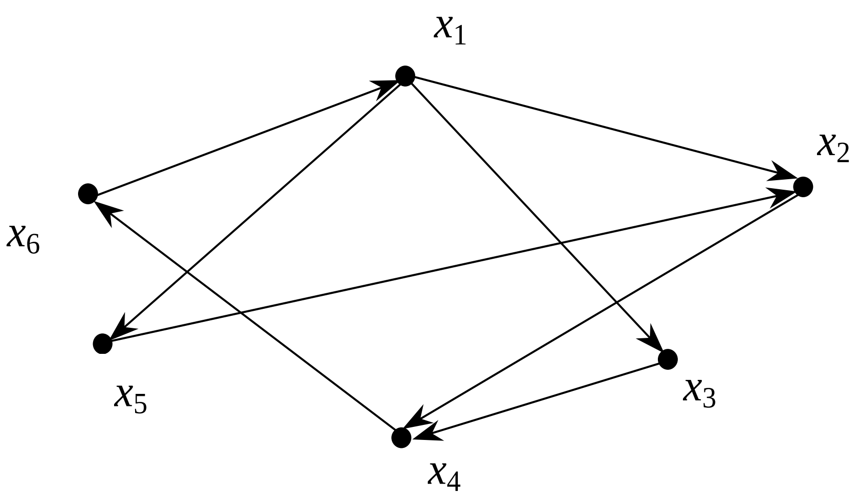


Рисунок 9 - Граф $G(X, \Gamma)$

Доминирующее множество является ***минимальным***, если нет другого доминирующего множества, содержащегося в нем.

Например, $\{x_1, x_4\}$ – минимальное доминирующее множество, а $\{x_1, x_4, x_6\}$ – нет.

Если P – семейство всех минимально доминирующих множеств, то число $\beta(G) = \min\limits_{S \in P} |S|$ называется ***числом доминирования***. В нашем случае $\beta(G) = 2$.

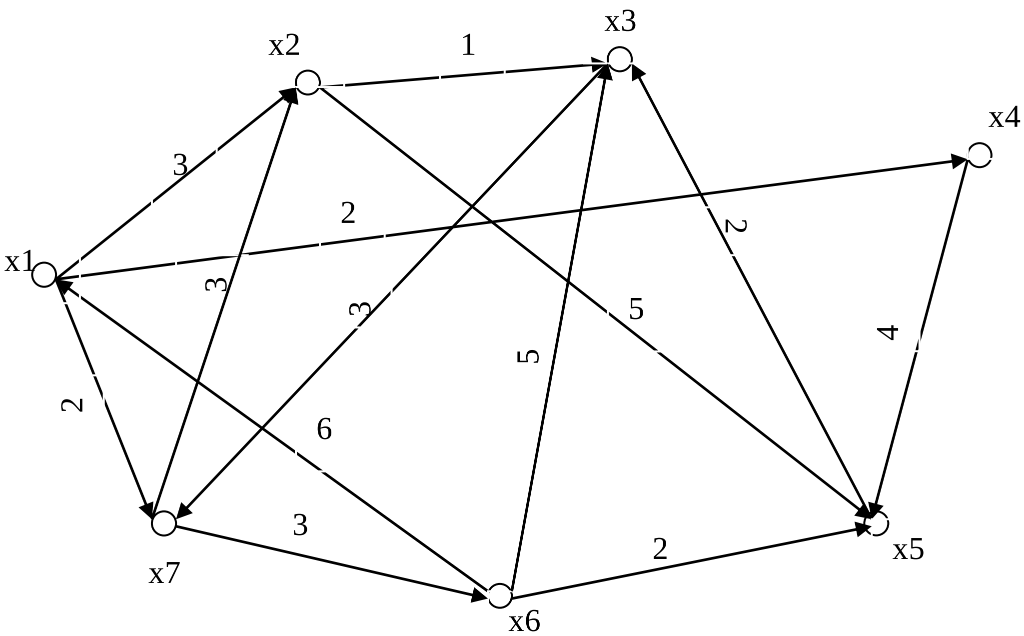
ПРИМЕР ВЫПОЛНЕНИЯ РАБОТЫ

Образец выполнения:

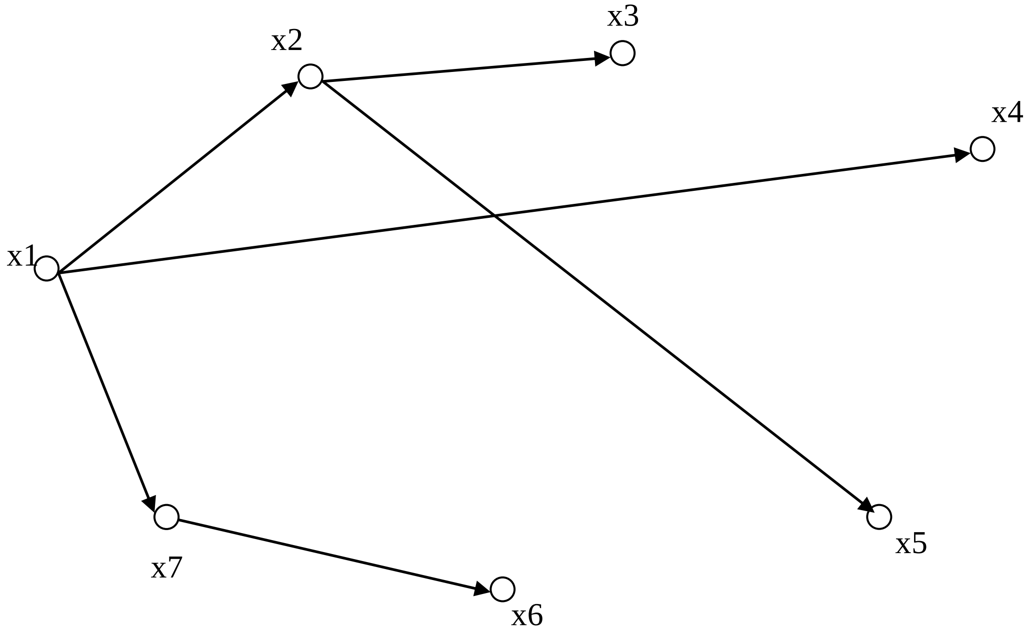
1)

j i	1	2	3	4	5	6	7
1		3		2			2
2			1		5		
3							3
4					4		
5			2				
6	6		5		2		
7		3				3	

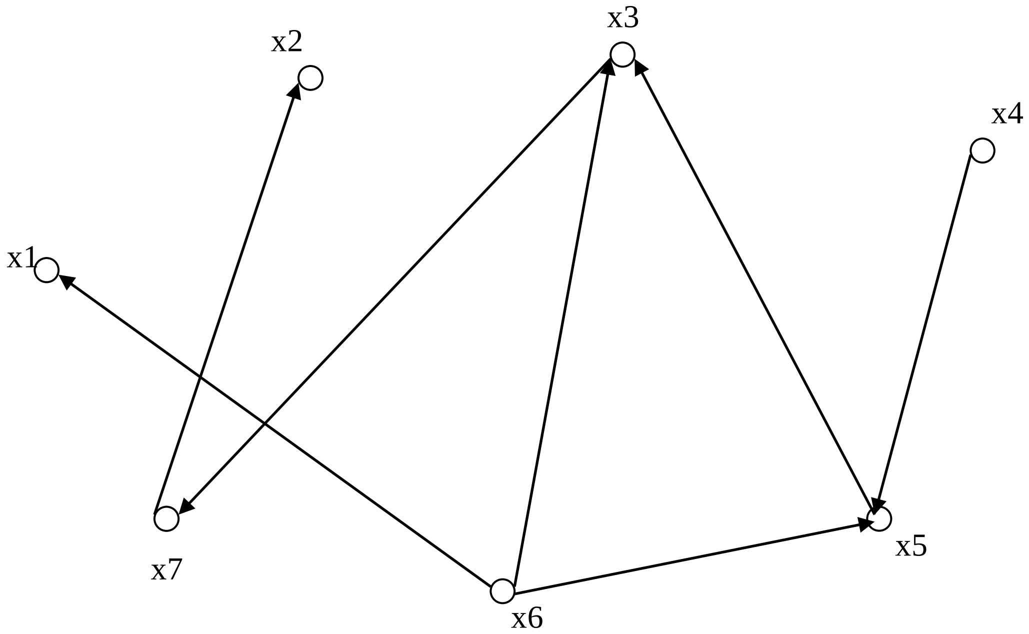
2)



3) Произвольное остовное дерево для заданного графа



Соответствующее ему ко-дерево



4)

1. Упорядочим рёбра в порядке возрастания их весов

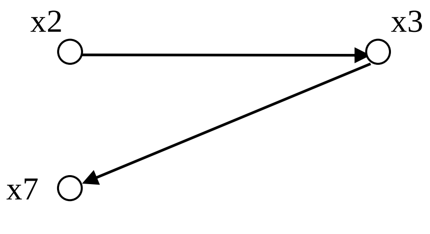
ребро	вес
a_{23}	2
a_{14}	3
a_{53}	3
a_{65}	3
a_{17}	4
a_{12}	4
a_{37}	4
a_{72}	5
a_{76}	6
a_{45}	6
a_{25}	6
a_{63}	6
a_{61}	7

2. Начиная с первого ребра в этом списке, добавляем рёбра в граф, соблюдая условие, при котором добавление нового ребра не должно приводить к образованию цикла. Добавляем ребра до тех пор, пока число рёбер в графе не станет равным n-1, где n-число вершин графа.

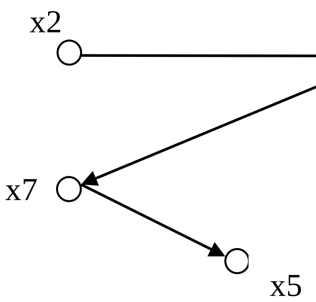
Шаг1

Корнем дерева является вершина x2.

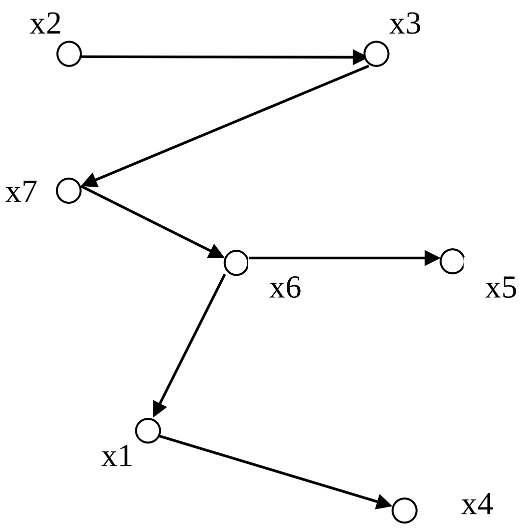
Шаг2



Шаг3



Шаг5



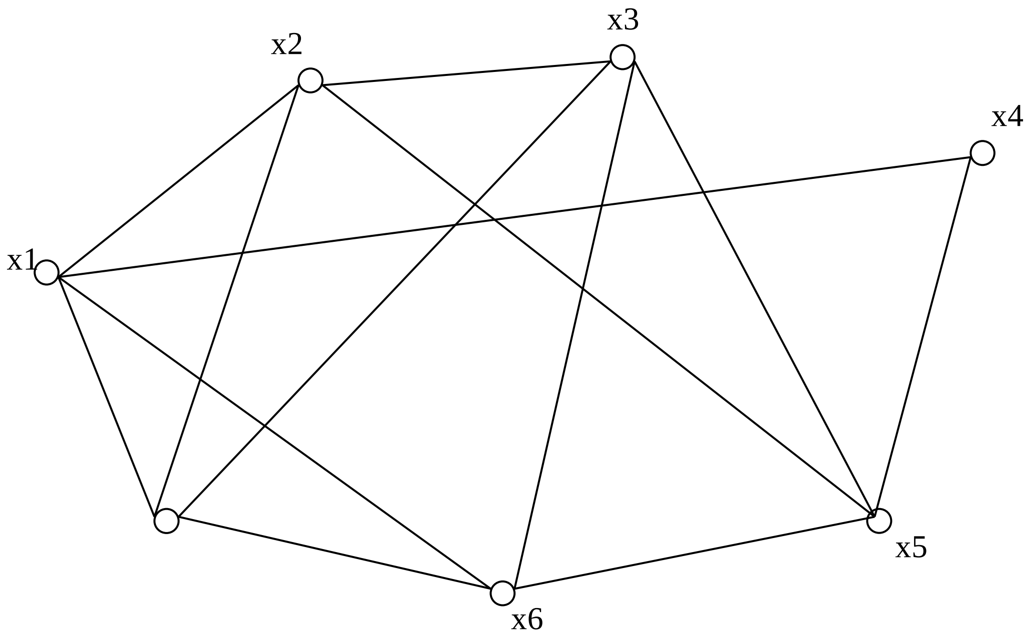
Минимальная длина остовного дерева *l*=17

5) Доминирующие множества: S = {x1, x2 }

S = {x1, x2,x7}

Число доминирования = 2

6) Представим наш граф как неорентированный



(x1, x5)
(x6, x2, x4)
(x3, x1)
(x7, x5)
(x7, x4)

⇒ Независимые множества

Число независимости равно 3.

Контрольные вопросы

1. Дать определение ориентированного дерева.
2. Как находится доминирующее множество вершин графа?
3. Что характеризует число доминирования?
4. Как формируется остовное дерево графа?
5. Как строится кодерево по отношению к остовному дереву?
6. Что характеризует число независимости графа?
7. Как определяется независимое множество вершин графа?
8. Для каких графов нельзя построить остовное дерево?
9. В чем состоит суть алгоритма Краскала?
10. Какой граф называется 3-деревом?
11. Какие требования предъявляются к вершине-лист дерева?
12. Какие требования предъявляются к вершине-корень дерева?
13. Сколько вершин и ребер содержит минимальное дерево?