

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
КАРАЧАЕВО-ЧЕРКЕССКИЙ ГОСУДАРСТВЕННЫЙ
УНИВЕРСИТЕТ ИМЕНИ У.Д. АЛИЕВА**

Х. Д. Шунгаров

ДИСКРЕТНАЯ МАТЕМАТИКА
(теория, решение задач, программирование)

ЧАСТЬ – II



УЧЕБНОЕ ПОСОБИЕ

Карачаевск – 2017

УДК-519.6
ББК 32.973.26-018.1

Печатается по решению редакционно-издательского
совета Карачаево-Черкесского государственного
университета имени У.Д. Алиева

Шунгаров Х.Д. Дискретная математика (теория, решение задач, программирование). Часть 2. / Учебное пособие. - Карачаевск: Изд-во КЧГУ, 2016. – 128 с.

ISBN 978-5-8307-0454-0

Представленное учебное пособие содержит теоретический материал и задания по основным темам дискретной математики, и предлагаются тестовые задания для проведения экзамена, практических занятий, также для самостоятельной работы. Учебное пособие составлено в соответствии с программой и предназначено для студентов и аспирантов, обучающихся по всем профилям подготовки бакалавров и магистров направлений «Информатика и вычислительная техника», «Прикладная информатика», «Прикладная математика и информатика» и ряда других специальностей, изучающих дискретную математику. Пособие может быть использовано студентами заочной формы обучения.

Составитель: Шунгаров Х.Д., к. физ.-мат. н., доцент

Рецензенты: *М.Х. Чанкаев*, к. физ.-мат. н., доцент

Р.А. Бостанов к. физ.-мат. н., доцент

ISBN 978-5-8307-0454-0

© Шунгаров Х.Д, 2017

© Карачаево-Черкесский государственный
университет имени У.Д. Алиева, 2017

СОДЕРЖАНИЕ

Введение.....	6
1. Введение в теорию графов.....	7
1.1. Основные понятия и определения.....	7
1.2. Подграфы и дополнения.....	10
1.3. Специальные графы.....	13
Задачи и упражнения.....	16
1.4. Матрицы графа.....	17
Задачи и упражнения.....	21
1.5. Степени вершин.....	23
Задачи и упражнения.....	25
1.6. Операции над графами.....	26
Задачи и упражнения.....	28
1.7. Маршруты, цепи, циклы.....	28
Задачи и упражнения.....	33
1.8. Связность и компоненты.....	33
1.9. Сильная связность.....	36
Задачи и упражнения.....	38
1.10. Матрицы достижимостей.....	39
1.11. Нахождение сильных компонент и построение конденсации.....	42
1.12. Базы и антибазы.....	45
1.13. Применение к исследованию структуры организаций.....	47
2. Деревья.....	48
Задачи и упражнения.....	50
2.1. Остовные деревья.....	51
2.2. Построение всех остовных деревьев графа.....	52
2.3. Алгоритм нахождения всех деревьев графа.....	54
2.4. Алгоритмы построения остовного дерева минимального веса графа.....	58
Задачи и упражнения.....	65
3. Метрические характеристики графа.....	67
3.1. Эксцентриситет, радиус и диаметр.....	67
Задачи и упражнения.....	68
3.2. Двудольные графы. Теорема Кенига.....	69
3.3. Алгоритм поиска в ширину.....	70
Задачи и упражнения.....	71
4. Эйлеровы графы.....	72

Задачи и упражнения	73
5. Гамильтоновы графы	74
Задачи и упражнения	75
6. Плоские и планарные графы	76
6.1. Определения	76
6.2. Грани плоского графа. Формула Эйлера	77
6.3. Критерий планарности графа	79
Задачи и упражнения	79
7. Нахождение кратчайших путей в графах	81
7.1. Постановка задачи	81
7.2. Алгоритм Дейкстры	81
7.3. Алгоритм Форда	87
7.4. Алгоритм Флойда нахождения кратчайших путей между всеми парами вершин	88
7.5. Применение к сетевому планированию и управлению.	90
Задачи и упражнения	91
8. Вершинная раскраска графа	92
8.1. Постановка задачи	92
8.2. Нижние оценки для $\chi(G)$	95
8.3. Верхние оценки для $\chi(G)$	95
8.4. Приближенные алгоритмы раскрашивания	96
9. Хроматические графы	98
10. Применения раскраски графа	103
10.1. Простая задача размещения (загрузки)	103
10.2. Составление графиков осмотра (проверки)	103
10.3. Задача распределения ресурсов	103
Задачи и упражнения	104
Комплект заданий для тестирования	105
Комплект вопросов к экзамену	119
Заключение	121
Список литературы	123

ПОЯСНИТЕЛЬНАЯ ЗАПИСКА

Целью пособия «дискретная математика» является освоение слушателями теоретических основ некоторых вводных тем важного раздела дискретной математики – теории графов с целью использования знаний в современных информационных технологиях, а также в прикладной научно-исследовательской и образовательной деятельности.

В современных условиях информатизации науки и образования, формирования глобального информационно-коммуникационного пространства к уровню квалификации бакалавров и магистров, а также научно-педагогических кадров предъявляются особые требования, которые, как правило, не обеспечиваются освоением базового курса информатики и информационных технологий. В связи с этим, изучение данного пособия ориентировано на:

- углубление общего информационного образования и информационной культуры будущих преподавателей и исследователей, ликвидацию возможных пробелов в усвоении базового курса прикладной информатики;

- овладение теорией дискретной математики и умение применять её для автоматизированного анализа научных данных;

- освоение прикладных технологий для решения разного рода практических задач с использованием современных методов программирования;

- изучение современных методов дискретной математики с использованием электронных средств поддержки образовательного процесса и приемов их интеграции с традиционными учебно-методическими материалами;

- формирование практических навыков решения научно-технических задач.

ВВЕДЕНИЕ

Дискретная математика - это раздел математики, главной спецификой которого является дискретность, т.е. антипод непрерывности. Это раздел математики, не связанный с понятиями бесконечности, предела и непрерывности.

В широком смысле дискретная математика включает в себя такие сложившиеся разделы математики как теория чисел, алгебра, математическая логика, комбинаторика и комбинаторный анализ и т.д. и ряд разделов, такие как, дискретная оптимизация, теория алгоритмов и анализ сложности задач, основы искусственного интеллекта, интеллектуальные системы, многокритериальная дискретная оптимизация, методы проектных решений и их анализ и т.д., которые наиболее интенсивно начали развиваться в середине XX века с внедрением ЭВМ.

Дискретная математика имеет широкий спектр приложений, прежде всего, в областях, связанных с информационными технологиями и компьютерами. При моделировании физических систем (электрических, экономических, агро - экологических и др.) удобной моделью является граф, каждая вершина которого соответствует функциональной или конструктивной компоненте, ребро - причинно-следственной связи. Графовая модель позволяет установить наглядную связь между структурой (топологией) системы и ее количественными (энергетическими, экономическими и др.) характеристиками.

В настоящее время интерес к дискретной математике неуклонно растёт. Всё больше в обязательную программу учебных заведений включаются курсы теории множеств, математической логики, комбинаторики, теории графов и их фрагменты. Специалисты в области современных компьютерных технологий уже осознали, что эти разделы математики являются фундаментом для построения необходимой сейчас хорошей теории математического обеспечения информационных технических систем. Многие специалисты, казалось бы, далёкие от математики, также начинают сознательно знакомиться с их содержанием.

1. ВВЕДЕНИЕ В ТЕОРИЮ ГРАФОВ

1.1. Основные понятия и определения

Пусть V - непустое множество, $V^{(2)}$ - множество всех его двухэлементных подмножеств. Пара $G=(V,E)$, где E — произвольное подмножество множества $V^{(2)}$, называется графом. Элементы множества V называются вершинами графа, а элементы множества E - ребрами. Итак, граф G - это конечное множество V вершин и множество E ребер, $E \subseteq V^{(2)}$. соответственно. Вершины и ребра графа G называются его элементами. Число $|V|$ вершин графа G называется его порядком и обозначается через $|G|$. Если $|G|=n, |E|=m$, то G называют (n,m) - графом. Если ребра $e \in E$ графа $G=(V,E)$ не имеют ориентации, т.е. определяются неупорядоченными парами вершин, то граф называется неориентированным (рис. 1.1(б)).

В данной работе рассматриваются только конечные графы, т.е. множество V предполагается конечным, хотя в определении графа конечность этого множества не требуется. Графически граф может быть представлен диаграммой, в которой вершина изображается точкой, а ребро - отрезком, соединяющим точки.

Например, если $V = \{v_1, v_2, v_3, v_4, v_5, v_6, v_7, v_8\}$,
 $E = \{e_1, e_2, e_3, e_4, e_5, e_6, e_7, e_8\}$, то граф $G=(V,E)$ представляется следующим образом:

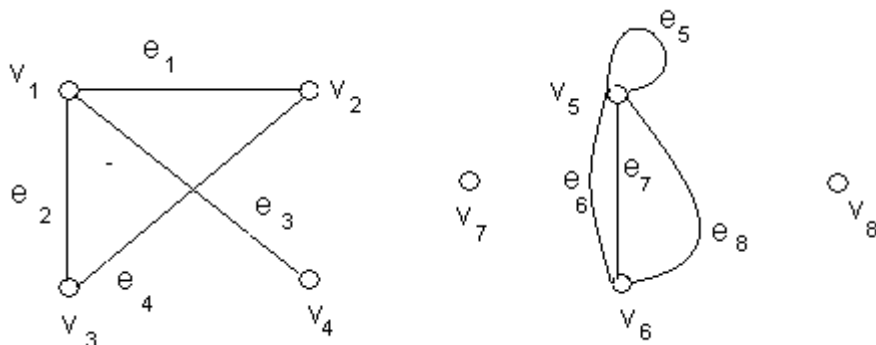


Рис. 1.1. Неориентированный граф $G=(V,E)$.

Ребро $e=(u,u)$ называется петлей. Говорят, что две вершины u,v графа смежны, если множество $e=(u,v)$ является ребром, и не смежны в противном случае. Если $e=(u,v)$ - ребро, то вершины u и

v называют его концами. В этой ситуации говорят также, что ребро e соединяет вершины u и v . Ребра с одинаковыми концевыми вершинами называются кратными. Число всех кратных ребер называется кратностью ребра. В графе на рис.1.2 ребра e_6, e_7, e_8 - кратные, e_5 - петля. Говорят, что ребро инцидентно своим концевым вершинам. Если два ребра имеют общую концевую вершину, то они называются смежными. Например, в графе на рис.1.2 ребро e_1 инцидентно вершинам v_1 и v_2 ; v_2 и v_3 являются смежными. Число ребер, инцидентных вершине v_i , называется степенью вершины и обозначается $d(v_i)$. Вершина $v_i: d(v_i)=1$ называется висячей. Вершина $v_i: d(v_i)=0$ называется изолированной. В графе G на рис.1 v_4 - висячая вершина, v_8 - изолированная, $d(v_6)=2$, ребро e_2 - висячее.

Вершина v и ребро e называются инцидентными, если v является концом ребра e (т. е. $e=(u,v)$), и не инцидентными в противном случае. Заметим, что смежность есть отношение между однородными элементами графа, тогда как инцидентность является отношением между разнородными элементами. Множество всех вершин графа G , смежных с некоторой вершиной v , называется окружением вершины v и обозначается через $N_G(v)$ или просто $N(v)$.

Ориентированное ребро v_1 графа $G=(V,A)$ определяется упорядоченной парой (v_1, v_2) , состоящей из начальной v_1 и конечной v_2 вершин и называется дугой. Направление дуги предполагается заданным от первой вершины ко второй. Так, например, на рис. 1.3(a) обозначение (v_1, v_2) относится к дуге a_1 , а (v_2, v_1) - к дуге a_2 .

Ориентированный граф (или орграф) - это пара (V,A) , где V - множество вершин, A - множество дуг, $A \subseteq V^{(2)}$. В случае, когда в орграфе $G=(V,A)$ мы хотим пренебречь направленностью дуг из множества A , то неориентированный граф, соответствующий G , будем обозначать как $G=(V,E)$.

Другое описание ориентированного графа G состоит в задании множества вершин V и соответствия Γ , которое показывает, как между собой связаны вершины.

Соответствие Γ называется отображением множества V в V , а граф в этом случае обозначается парой $G=(V, \Gamma)$. Для графа на рис. 1.3(а) имеем $\tilde{A}(v_1)=\{v_2, v_5\}$, т. е. вершины v_2 , v_5 являются конечными вершинами дуг, у которых начальной вершиной является v_1 . $\Gamma(v_2)=\{v_1, v_3\}$, $\Gamma(v_3)=\{v_1\}$, $\Gamma(v_4)=\emptyset$, $\Gamma(v_5)=\{v_4\}$.

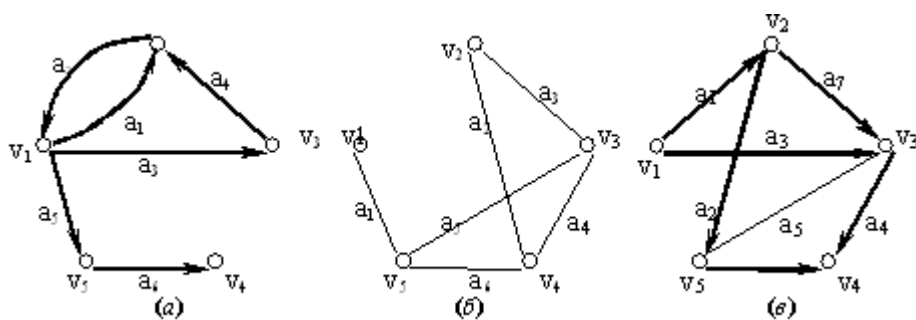


Рис. 1.2. (а) Орграф, (б) Неориентированный граф, (в) Смешанный граф.

В случае неориентированного графа или смешанного графа, содержащего и дуги, и неориентированные ребра (рис. 1.2(б) и 1.2(в)), предполагается, что соответствие Γ задает такой эквивалентный ориентированный граф, который получается из исходного графа заменой каждого неориентированного ребра двумя противоположно направленными дугами, соединяющими те же самые вершины. Так, например, для графа, приведенного на рис. 1.2(б), имеем $\tilde{A}(v_5)=\{v_1, v_3, v_4\}$, $\tilde{A}(v_1)=\{v_5\}$ и т. д.

Поскольку $\tilde{A}(v_i)$ представляет собой множество таких вершин $v_j \in V$, для которых в графе G существует дуга (v_i, v_j) , то через $\Gamma^{-1}(v_i)$ естественно обозначить множество вершин v_k , для которых в G существует дуга (v_k, v_i) . Отношение $\Gamma^{-1}(v_i)$ принято называть обратным соответствием. Для графа, изображенного на рис. 1.2(а), имеем

$$\tilde{A}^{-1}(v_1) = \{v_2, v_3\},$$

$$\tilde{A}^{-1}(v_2) = \{v_1\}$$

и т. д. Вполне очевидно, что для неориентированного графа $\tilde{A}^{-1}(v_i) = \tilde{A}(v_i)$ для всех $v_i \in X$. Когда отображение Γ действует не на одну вершину, а на множество вершин $V_q = \{v_1, v_2, \dots, v_q\}$, то под $\tilde{A}(V_q)$

понимают объединение $\tilde{A}(v_1) \cup \tilde{A}(v_2) \cup \dots \cup \tilde{A}(v_q)$, т. е. $\tilde{A}(V_q)$ является множеством таких вершин $v_i \in V$, что для каждой из них существует дуга (v_i, v_j) в G , где $v_j \in V$. Для графа, приведенного на рис. 1.3(a), $\tilde{A}(\{v_2, v_5\}) = \{v_1, v_3, v_4\}$ и $\tilde{A}(\{v_1, v_3\}) = \{v_2, v_5, v_1\}$.

Отображение $\tilde{A}(\tilde{A}(v_i))$ записывается как $\tilde{A}^2(v_i)$. Аналогично «тройное» отображение $\tilde{A}(\tilde{A}(\tilde{A}(v_i)))$ записывается как $\tilde{A}^3(v_i)$ и т. д.

Для графа, показанного на рис. 1.2(a), имеем:
 $\Gamma^2(v_1) = \Gamma(\Gamma(v_1)) = \Gamma(\{v_2, v_5\}) = \{v_1, v_3, v_4\}$,

$\Gamma^3(v_1) = \Gamma(\Gamma^2(v_1)) = \Gamma(\{v_1, v_3, v_4\}) = \{v_1, v_2, v_5\}$

и т. д. Аналогично понимаются обозначения $\tilde{A}^{-2}(v_i), \tilde{A}^{-3}(v_i)$.

1.2. Подграфы и дополнения

Граф $G' = (V', E')$ называется подграфом (или частью) графа $G = (V, E)$, если $V' \subseteq V, E' \subseteq E$. Если G' - подграф графа G , то говорят, что G' содержится в G . Подграф G' называется остовным подграфом (или фактором), если $V' = V$.

Если V' - множество вершин подграфа G' , а множество E' его ребер совпадает с множеством всех таких ребер графа $G = (V, E)$, оба конца которых принадлежат E' , то G' называется вершинно-порождённым подграфом или порожденным (индуцированным) множеством V' , и обозначается через $\langle V' \rangle$.

Если подграф $G' = (V', E')$ не содержит изолированных вершин, то каждая вершина является концевой некоторого ребра E . Тогда E однозначно определяет V' и, следовательно, подграф G' . Подграф $G' = (V', E')$ называется рёберно-порожденным и обозначается через $\langle E' \rangle$.

Пусть дан орграф $G = (X, A)$. Остовный подграф G' орграфа G определяется аналогично.

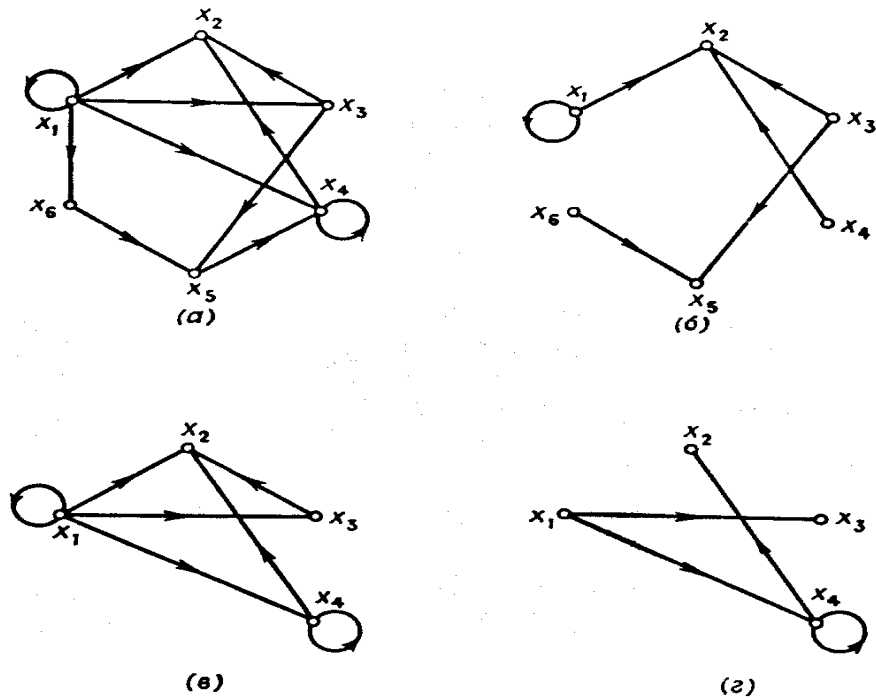


Рис. 1.3. (а) Граф, (б) Остовный подграф, (в) Порожденный подграф, (г) Подграф.

Вершинно - порожденным подграфом G' орграфа $G=(X, \Gamma, A)$ называется подграф $G'=(X', \Gamma')$, для которого $X' \subseteq X$ и для каждой вершины $x_i \in X', \Gamma'(x_i) = \Gamma(x_i) \cap X'$.

Таким образом, порожденный подграф $\langle X' \rangle$ орграфа $G=(X, A)$ состоит из подмножества X' множества вершин исходного графа и всех таких дуг графа G , у которых конечные и начальные вершины принадлежат подмножеству X' . На рис. 1.3(в) показан порожденный подграф графа, приведенного на рис. 1.3(а), содержащий только вершины x_1, x_2, x_3, x_4 и дуги, которые их связывают. Граф, изображенный на рис. 1.3(г), является подграфом графа, приведенного на рис. 1.3(а).

Пусть p - некоторое свойство, которым граф может обладать.

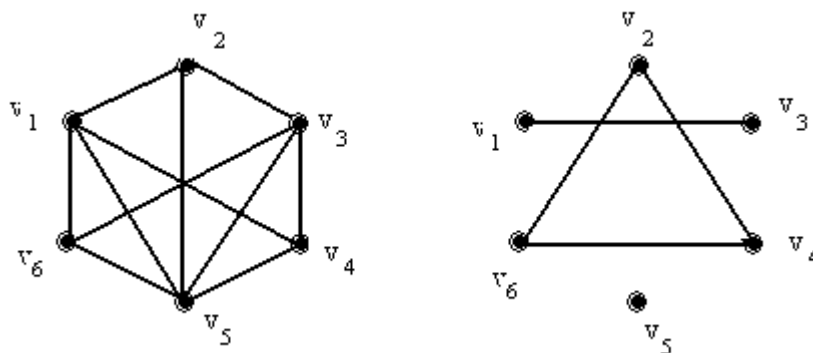
Максимальным подграфом графа G относительно свойства p называется подграф $G'=(V', E')$ графа G , обладающий этим свойством и такой, что не существует другого собственного подграфа G'' , и который также обладает свойством P .

Минимальным подграфом графа G относительно свойства p называется подграф $G'=(V', E')$ графа G , обладающего свойством p , и не существует другого подграфа G'' , который обладает свойством p и не является собственным подграфом графа G' .

Множество рёбер вершинно - порождённого подграфа $\langle V' \rangle$ на множестве V' является таким *максимальным подмножеством* E , что концевые вершины всех его рёбер принадлежат V' .

Множество вершин V' рёберно - порождённого подграфа $\langle E' \rangle$ является таким *минимальным подмножеством* V , содержащим все концевые вершины рёбер в E' .

Граф $\overline{G} = (V, E')$ называется дополнением простого графа $G = (V, E)$, если ребро $(v_i, v_j) \in E'$ тогда и только тогда, когда $(v_i, v_j) \notin E$.



а) Граф $G = (V, E)$. б) Дополнение \overline{G} .

Рис.1.4. Граф $G = (V, E)$ и его дополнение.

С дополнением графа связана так называемая задача о рукопожатиях.

Лемма 1.1. В любой группе из 6 человек трое либо попарно знакомы, либо попарно не знакомы.

Доказательство. Поставим в соответствие людям вершины, а наличие знакомства между ними - рёбра, соединяющие соответствующие вершины. Получим соответствующий простой 6-вершинный граф. Используя определение дополнения графа, получим утверждение, эквивалентное лемме 1.1.

Лемма 1.2. Для любого простого графа $G = (V, E)$ с шестью вершинами верно, что G или \overline{G} содержат три попарно- смежные вершины.

Доказательство. Рассмотрим произвольную вершину графа $G: |V|=6$. Пусть вершина $v \in V$ не смежна с тремя вершинами в G , тогда v смежна с тремя вершинами в \overline{G} . Не нарушая общности предположим, что v смежна с некоторыми вершинами

$v_1, v_2, v_3 \in \bar{G}$. Если две из этих вершин, например, v_1, v_2 смежны в \bar{G} , то вершины v_1, v_2, v_3 попарно смежны в \bar{G} и лемма 1.2 доказана. Если в G никакие две из этих трёх вершин не смежны, то они попарно смежны в G и лемма 1.2 доказана.

1.3. Специальные графы

Если любые две вершины графа G смежны, то граф называется полным. Полный неориентированный граф, построенный на n вершинах, обозначается через K_n .

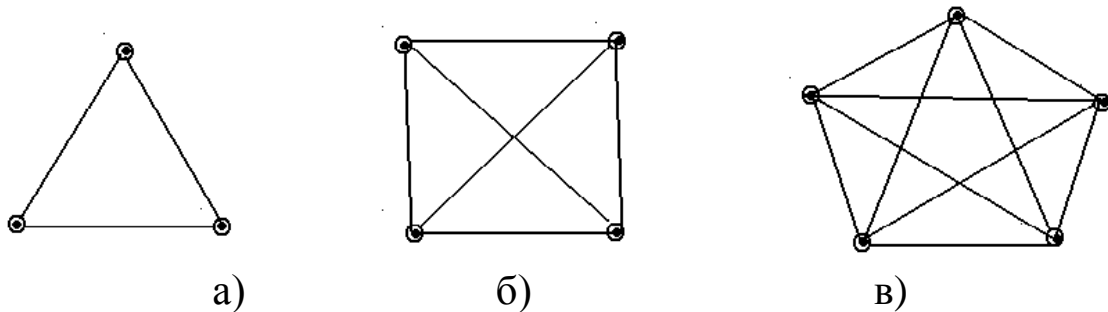


Рис.1.5. а)- граф K_3 , б) - граф K_4 , в) - граф K_5 .

Из определения дополнения графа следует, что $\bar{G} \cup G = K_n$.

Граф $G=(V,E)$ называется двудольным, если существует такое разбиение множества V его вершин на два подмножества (доли) V^1 и V^2 , что концы каждого ребра принадлежат либо V^1 либо V^2 .

Орграф $G=(V,A)$ называется двудольным, если его неориентированный двойник $G=(V,E)$ - двудольный граф и обозначается $G=(V^1, V^2, A)$.

Двудольный орграф $G=(V^1, V^2, A)$ называют полным, если для любых двух вершин $x_i \in V^1$ и $x_j \in V^2$ существует дуга $(v_i, v_j) \in A$.

Если $|V^1|=n$, $|V^2|=m$, то полный двудольный граф $G=(V^1, V^2, E)$ обозначается через $K_{m,n}$. При $m=1$ получаем звезду $K_{1,n}$. Очевидно, что $K_{1,1} = K_2 = P_2, K_{1,2} = P_3, K_{2,2} = C_4$.

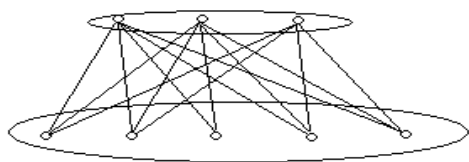


Рис.1.6. Граф $K_{3,5}$.

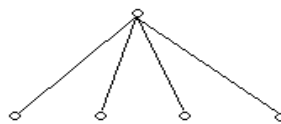


Рис.1.7. Звезда $K_{1,5}$.

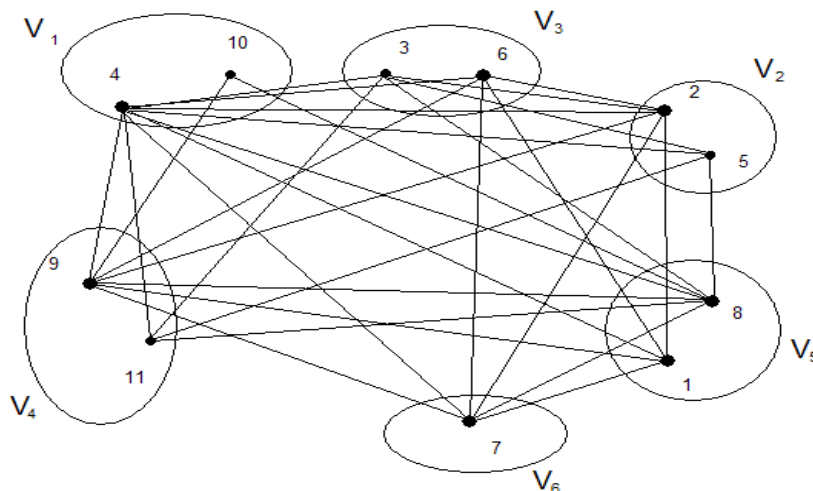


Рис.1.8. k -дольный граф, $k=6$.

Аналогично определяются k -дольный граф для $k = 3, 4, \dots$

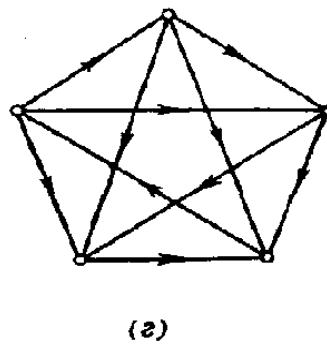
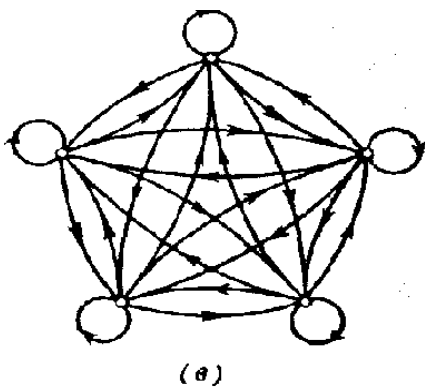
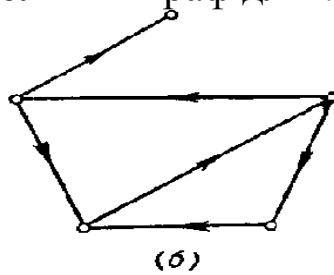
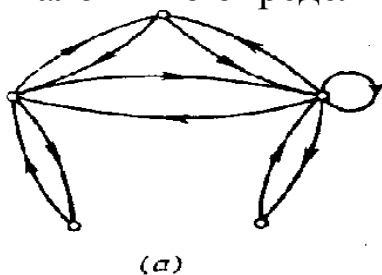


Рис. 1.9 (а)- Симметрический граф, (б)- анти симметрический граф, (в)- полный симметрический граф, (г)- полный антисимметрический граф.

Граф $G=(V,A)$ называется симметрическим (рис. 1.9(а)), если в множестве дуг A для любой дуги (v_i, v_j) существует также противоположно ориентированная дуга (v_j, v_i) .

Антисимметрическим графом (рис. 1.9(б)) называется такой граф, для которого справедливо следующее условие: если $(v_i, v_j) \in A$ то в множестве A нет противоположно ориентированной дуги, т. е. $(x_j, x_i) \notin A$. Очевидно, что в антисимметрическом графе нет петель.

Рассмотрим следующий пример: множество вершин графа представляет группу людей, дуга, направленная от вершины v_i к вершине v_j , означает, что v_i является другом или родственником v_j , тогда данный граф должен быть симметрическим. С другой стороны если дуга, направленная от v_i к v_j означает, что вершина v_i подчинена вершине v_j , то такой граф должен быть антисимметрическим.

Полный симметрический граф называют также турниром.

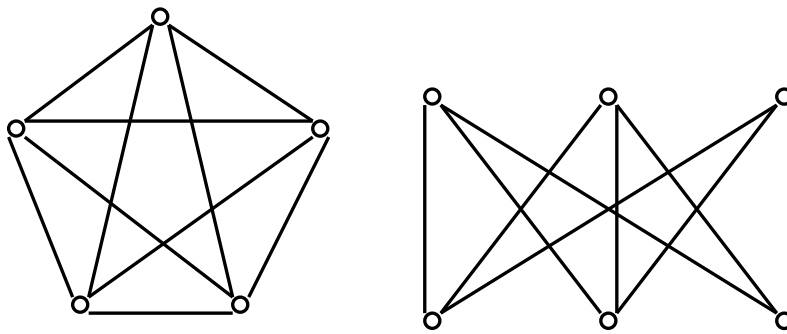


Рис. 1.10. Непланарные графы: (а) - K_5 , (б) - $K_{3,3}$.

Пример 1.1. Являются ли полными (без учета петель) графы G_1 , изображенные на рис. 2.2 и 2.3?

Решение. Граф G_1 не является полным, т.к. не все пары его вершин соединены ребрами. Например, (v_1, v_3) , (v_3, v_4) и другие.

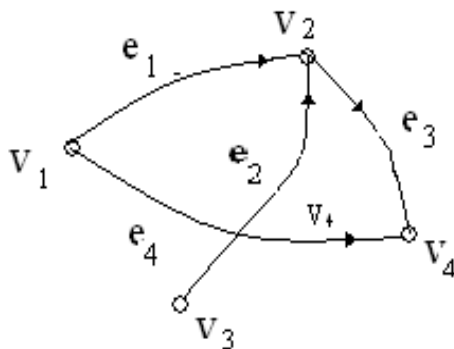


Рис. 1.11. Граф G_1 .

Задачи и упражнения

1. Среди графов, указанных на рис. 1.12, выделить полные графы (без учета петель).

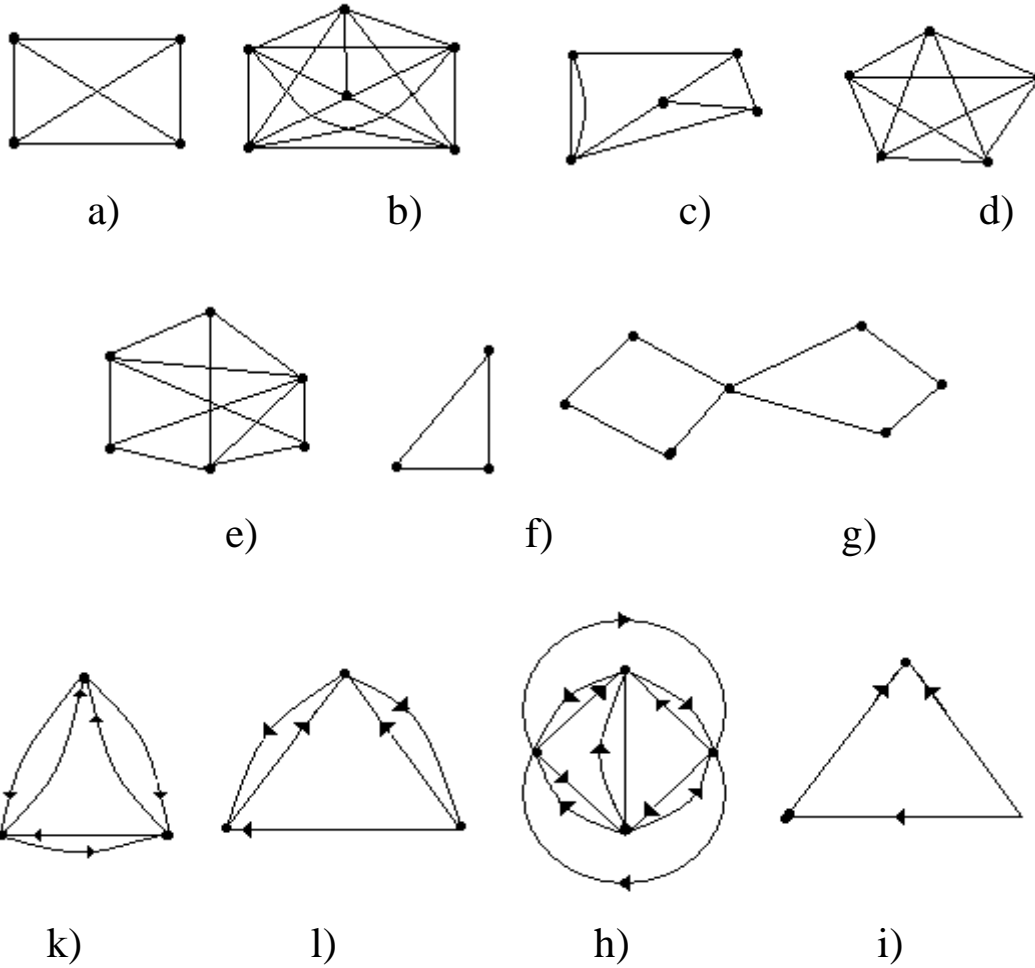


Рис.1.12. Типы графов.

1.4. Матрицы графа

1. Матрица смежности

Матрица, каждый элемент которой равен 0 или 1, называется *бинарной*.

Пусть дан граф G , его *матрица смежности* обозначается через $A=[a_{ij}]$ и определяется следующим образом:

$$a_{ij} = \begin{cases} 1, & \text{если дуга } (v_i, v_j) \in A \\ 0 & \text{в противном случае} \end{cases}, \text{ где } A - \text{множество дуг орграфа}$$

Матрица смежности графа G на рис. 1.13, имеет вид

	v_1	v_2	v_3	v_4	v_5	v_6
v_1	0	1	1	0	0	0
v_2	0	1	0	0	1	0
$A = v_3$	0	0	0	0	0	0
v_4	0	0	1	0	0	0
v_5	1	0	0	1	0	0
v_6	1	0	0	0	1	1

Матрица смежности полностью определяет структуру графа. Например, сумма всех элементов строки v_i матрицы дает полустепень исхода вершины v_i , а сумма элементов столбца v_i - полустепень захода вершины v_i . Множество столбцов, имеющих 1 в строке v_i , есть множество $\Gamma(v_i)$, а множество строк, которые имеют 1 столбце v_i , совпадает с множеством $\Gamma^{-1}(v_i)$. Это симметрическая матрица с нулями на диагонали. Число единиц в строке равно степени соответствующей вершины.

Возведем матрицу смежности в квадрат. Пусть элемент $a_{ik}^{(2)}$ матрицы A^2 определяется по формуле

$$a_{ik}^{(2)} = \sum_{j=1}^n a_{ij} a_{jk} \quad (1.1)$$

Слагаемое в уравнении (1.1) равно 1 тогда и только тогда, когда оба числа a_{ij} и a_{jk} равны 1, в противном случае оно равно 0. Поскольку

из равенств $a_{ij} = a_{jk} = 1$ следует существование пути длины 2 из вершины v_i к вершине v_k , проходящего через вершину v_j , то $a_{ik}^{(2)}$ равно числу путей длины 2, идущих из v_i в v_k .

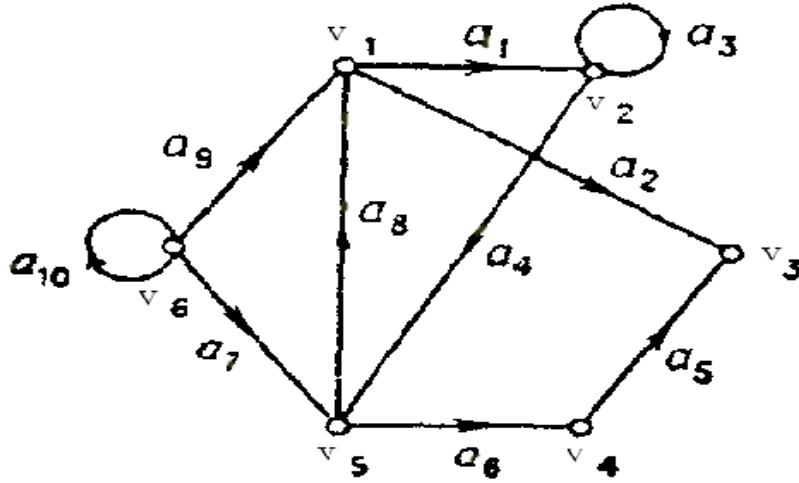


Рис. 1.13

Аналогично если $a_{ik}^{(p)}$ является элементом матрицы $A^{(p)}$, то $a_{ik}^{(p)}$ равно числу путей (не обязательно орцепей или простых орцепей) длины p , идущих от v_i к v_k .

2. Матрица инциденций

Определим матрицу инцидентности графа G .

Пусть $G - (n, m)$ - граф, $V = \{v_1, v_2, \dots, v_n\}$, $E = \{e_1, e_2, \dots, e_m\}$.

Бинарная $m \times n$ матрица $B = B(G)$, для которой выполняются условия

$$b_{k,l} = \begin{cases} 1, & \text{если вершина } v_k \text{ и ребро } e_l \text{ инцидентны,} \\ 0 & \text{в противном случае.} \end{cases}$$

называется матрицей инцидентности графа G .

В каждом ее столбце ровно две единицы, равных столбцов нет. Соответствие $G \rightarrow B(G)$ является биекцией множества помеченных (n, m) - графов с занумерованными ребрами на множестве $m \times n$ - матриц, удовлетворяющих описанным условиям. Для ориентированных графов матрица инцидентности имеет вид:

$$b_{k,l} = \begin{cases} 1, & \text{если вершина } v_k \text{ является началом дуги } a_l \\ -1, & \text{если вершина } v_k \text{ является концом дуги } a_l \\ 0, & \text{если вершина } v_k \text{ и дуга } a_l \text{ не инцидентны} \end{cases}$$

Для графа, приведенного на рис. 1.13, матрица инциденций имеет вид

	a ₁	a ₂	a ₃	a ₄	a ₅	a ₆	a ₇	a ₈	a ₉	a ₁₀
v ₁	1	1	0	0	0	0	0	-1	-1	0
v ₂	-1	0	0	1	0	0	0	0	0	0
v ₃	0	-1	0	0	-1	0	0	0	0	0
v ₄	0	0	0	0	1	-1	0	0	0	0
v ₅	0	0	0	-1	0	1	-1	1	0	0
v ₆	0	0	0	0	0	0	1	0	1	0

Поскольку каждая дуга инцидентна двум различным вершинам, за исключением того случая, когда дуга образует петлю, то каждый столбец либо содержит один элемент, равный 1, и один равный -1, либо все элементы столбца равны 0.

Пример 1.2. Дан ориентированный граф (рис. 1.13). Построить его матрицы смежности и инцидентности.

Решение. В соответствии с определением матрица смежности есть квадратная матрица с элементами множества вершин в качестве координат ее столбцов и строк.

Элемент матрицы в строке i и столбце j равен 1, если вершина i смежна вершине j , -1 - если вершина i смежна вершине j и 0 - если вершины i и j не смежны. Матрица смежности приведена в таблице 1.1. В матрице инцидентности координатами строк являются элементы множества вершин, а координатами столбцов - элементы множества ребер.

Таблица 1.1.

V \ X	a	b	c	d
a	0	1	0	1
b	-1	0	-1	1
c	0	1	0	0
d	-1	-1	0	0

Таблица 1.2.

V \ X	V ₁	V ₂	V ₃	V ₄
a	1	0	0	1
b	-1	-1	1	0
c	0	1	0	0
d	0	0	-1	-1

Элемент матрицы в строке i и столбце j равен 1, если ребро j исходит из вершины i , -1 - если ребро j входит в вершину i , 0 -

если ребро j не инцидентно вершине i . Матрица инцидентности приведена в таблице 1.2.

Пример 1.3. На рис. 1.14 задан граф G . Построить матрицу смежности и выяснить, сколько путей длины три существует в графе G .

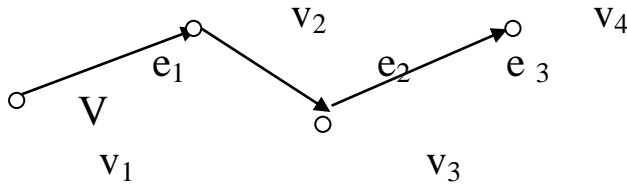


Рис. 1.14.

Решение.

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$A^2 = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$A^3 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

$$A^4 = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Элемент $a_{14}^{(3)} = 1$, следовательно, в данном графе существует единственный путь длиной три. Это путь из вершины v_1 в вершину

$$v_4: v_1 \xrightarrow{e_1} v_2 \xrightarrow{e_2} v_3 \xrightarrow{e_3} v_4.$$

Все элементы матрицы A^4 равны нулю, следовательно, в графе отсутствуют пути длины четыре.

Два графа G_1 и G_2 называются изоморфными, если существует такое взаимно-однозначное соответствие между множествами их вершин и рёбер, что соответствующие рёбра графов инцидентны соответствующим вершинам. Определение изоморфизма запишем короче:

$$G_1 \cong G_2 \Rightarrow \exists f : (V_1 \rightarrow V_2) \wedge (E_1 \rightarrow E_2) : v_i, v_j \in V_1 \rightarrow v'_i, v'_j \in V_2 \Leftrightarrow (v_i, v_j) = e \rightarrow e' = (v'_i, v'_j)$$

Пример 1.4 Изоморфны ли графы, изображенные на рис. 1.14 и 1.15?

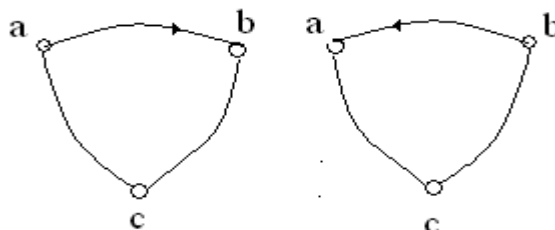


Рис.1.15.

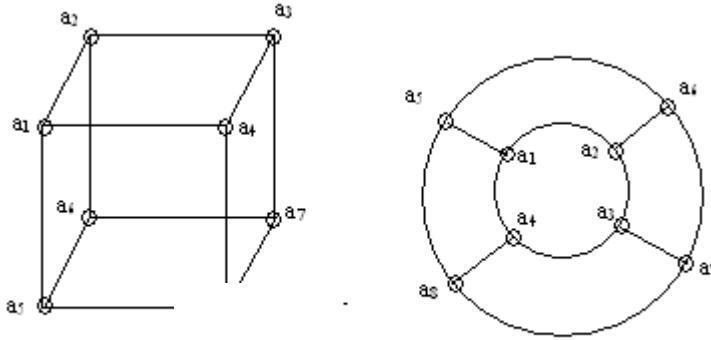


Рис. 1.16.

Решение. Графы G_1 и G_2 не изоморфны, хотя они и имеют одинаковое число вершин и ребер. Но в графе A_1 одно из ребер направлено от a к b , а в графе A_2 оно направлено в другую сторону. Графы B_1 и B_2 изоморфны, т.к. они имеют одно и то же число вершин и любые две вершины графа B_1 соединены ребром только тогда, когда соответствующие им вершины графа B_2 также соединены ребром.

Задачи и упражнения

1. Построить графы, матрицы смежности которых указаны:

$$M_1 = \begin{pmatrix} 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{pmatrix}; \quad M_2 = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}; \quad M_3 = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}; \quad M_4 = \begin{pmatrix} 0 & 1 & -1 \\ -1 & 0 & 1 \\ 1 & -1 & 0 \end{pmatrix};$$

$$M_5 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}; \quad M_6 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}; \quad M_7 = \begin{pmatrix} 0 & 1 & 0 & 0 & 1 \\ -1 & 0 & -1 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ -1 & -1 & 0 & -1 & 1 \end{pmatrix};$$

$$M_8 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ -1 & 0 & -1 & 1 \\ 0 & 1 & 0 & 0 \\ -1 & -1 & 0 & 0 \end{pmatrix}; \quad M_9 = \begin{pmatrix} 1 & 1 & -1 \\ -1 & 1 & 1 \\ 1 & -1 & 1 \end{pmatrix}.$$

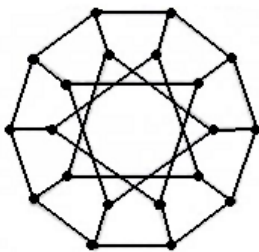
2. Построить графы, матрицы инцидентности которых указаны:

$$M_1 = \begin{pmatrix} 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{pmatrix}; \quad M_2 = \begin{pmatrix} -1 & 1 & 0 & 0 \\ 0 & -1 & 1 & 0 \\ 0 & 0 & -1 & 1 \\ 1 & 0 & 0 & -1 \end{pmatrix}; \quad M_3 = \begin{pmatrix} 1 & 0 & 1 & -1 & 0 \\ -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & -1 & 0 & -1 \\ 0 & 0 & 0 & 0 & 1 \end{pmatrix};$$

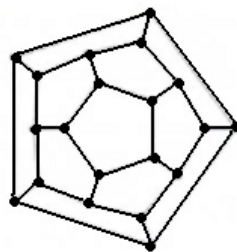
$$M_4 = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ -1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 \end{pmatrix}; \quad M_5 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ -1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & -1 & 0 & 1 & 1 \\ 0 & -1 & -1 & -1 & 0 & -1 \end{pmatrix};$$

$$M_6 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \end{pmatrix}; \quad M_7 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 1 \\ 0 & 0 & -1 & -1 & 0 \\ -1 & 0 & 0 & 1 & -1 \end{pmatrix}; \quad M_8 = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \end{pmatrix}.$$

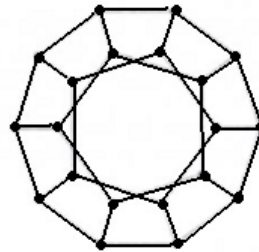
3. Верно ли, что два графа изоморфны, если
- у них по 10 вершин, степень каждой из которых равна 9?
 - у них по 8 вершин, степень каждой из которых равна 3?
 - они связны, без циклов и содержат по 6 ребер?
4. Найдите все попарно неизоморфные графы со степенями вершин (3,3,3,3,4,4).
5. Все степени вершин n -вершинного графа равны 2. Сколько среди таких графов попарно неизоморфных?
- Вычислите ответ для $3 < n < 10$;
 - Найдите рекуррентное соотношение, позволяющее вычислить ответ для произвольного n . (Указание: вспомните числа Стирлинга.)
6. Найдите все попарно неизоморфные графы со степенями вершин: а) (3,3,3,3,3,3); б) (6,6,6,6,6,6,6,6).
7. Какие пары из приведенных ниже графов изоморфны:



a)



b)



c)

8. Дана $(0, 1)$ -матрица $C_{n \times n}$ смежности графа. Составить программу изображения этого графа.

1.5. Степени вершин

Степенью (или *валентностью*) *вершины графа* G называется число инцидентных ей ребер, т. е. число вершин в ее окружении. Граф G называется *однородным* или *регулярным* степени r , если степени всех его вершин равны между собой и равны r .

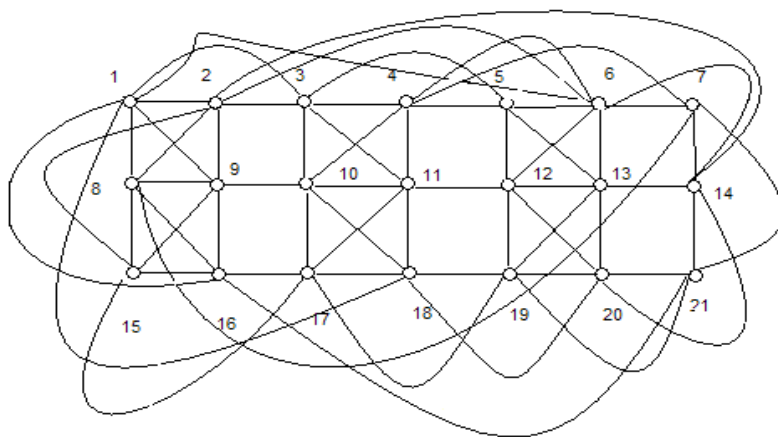


Рис.1.17. Регулярный граф степени $r=6$.

Будем обозначать степень вершины v через $\deg v$. Тем самым ($\deg v == |N(v)|$).

Максимальная и минимальная степени вершин графа G обозначаются символами: $\Delta(G) = \max_{v \in VG} \deg v$, $\delta(G) = \min_{v \in VG} \deg v$.

Неориентированный граф $G=(V,E)$, без петель и кратных ребер называется простым.

Теорема 1.1. Пусть задан граф $G=(V,E)$, где $|V|=n$ – число вершин, $|E|=m$ – число ребер. Сумма степеней всех вершин графа G

равна $2 \cdot m$, $\left(\sum_{i=1}^n \deg v_i = 2m \right)$.

Доказательство. Поскольку каждое ребро инцидентно двум вершинам, оно добавляет двойку к сумме степеней этого графа. Поэтому суммарное число всех равно $2 \cdot m$. Теорема 1.1 доказана.

Теорема 1.2. Число вершин нечётной степени в любом графе чётно.

Доказательство. Пусть число вершин в графе G равно $n = |V|$. Не нарушая общности предположим, что степени первых s вершин v_1, v_2, \dots, v_s - чётны, а степени оставшихся $n-s$ вершин $v_{s+1}, v_{s+2}, \dots, v_n$ нечётны. Тогда можно записать

$$\sum_{i=1}^n \deg v_i = \sum_{i=1}^s \deg v_i + \sum_{i=s+1}^n \deg v_i$$

Из теоремы 1.1 следует, что сумма в левой части равенства - чётное число, первая сумма в правой части - чётное число, поскольку каждый член этой суммы также является чётным числом. Отсюда следует, что вторая сумма должна быть чётным числом. Однако, это число состоит из нечётных слагаемых чисел. Поэтому количество таких чисел должно быть чётным числом. Теорема 1.2. доказана.

Список степеней вершин графа называется его *степенной последовательностью*. Порядок членов в этой последовательности роли не играет. Вершина степени 0 называется *изолированной*, вершина степени 1 - *концевой* (или *висячей*). Ребро, инцидентное концевой вершине, также называется *концевым*. Вершина графа, смежная с каждой другой его вершиной называется *доминирующей*.

Число дуг, которые имеют вершину v_i своей начальной вершиной, называется *полустепенью исхода* вершины v_i , и, аналогично, число дуг, которые имеют v_i своей конечной вершиной, называется *полустепенью захода* вершины v_i . Таким образом, на рис. 1.13 полустепень исхода вершины x_6 , обозначаемая через $d_0(v_6)$, равна $|\Gamma(v_6)|=2$, и полустепень захода вершины v_6 , обозначаемая через $d_t(v_6)$, равна $|\Gamma^{-1}(v_6)|=1$.

Совершенно очевидно, что сумма полустепеней захода всех вершин графа, а также сумма полустепеней исхода всех вершин равны общему числу дуг графа G , т. е.

$$\sum_{i=1}^n d_0(v_i) = \sum_{i=1}^n d_t(v_i) = m, \quad (1.2)$$

где n — число вершин и m — число дуг графа G .

Для неориентированного графа $G=(X, \Gamma)$ *степень* вершины x_i определяется аналогично - с помощью соотношения $d(v_i) \equiv |\tilde{A}(v_i)|$, и

когда не может возникнуть недоразумений, мы будем обозначать степень вершины v_i через d_i .

Задачи и упражнения

1. а) В графе n вершин. Сколько в нем может быть ребер? б) В графе 25 ребер. Сколько в нем может быть вершин?
2. Существует ли граф на 8 вершинах, в котором 23 ребра и есть вершина степени 1?
3. Привести пример графа, имеющего n вершин и $(n - 1)(n - 2)/2$ ребер.
4. Можно ли нарисовать на плоскости 9 отрезков так, чтобы каждый пересекался ровно с тремя другими?
5. Матрица A размера $n \times n$ состоит из 0 и 1. Известно, что на главной диагонали стоят нули $a_{ii}=0$, а матрица симметрична ($a_{ij} = a_{ji}$). Докажите, что количество строк матрицы A , в которых находится нечетное количество единиц, четно.
5. Докажите, что сумма степеней вершин равна удвоенному числу ребер.
6. Приведите пример графа (или докажите, что его не существует), степени вершин которого равны
 - а) (1,1,1,2,3,3,4);
 - б) (1,1,1,2,3,3,3):
 - и) (1,1,3,3,3,3,6);
 - г) (5,5,1,7,7,7,7);
 - д) (4,4,1,4,4,5,5):
 - е) (1,1,2,2,2,6,6).
7. Может ли в государстве, в котором из каждого города выходит 3 дороги, быть ровно 100 дорог?
8. Докажите, что в графе существуют, по крайней мере, 2 вершины, степени которых равны.
9. Докажите, что не существует многогранника, у которого было бы ровно семь ребер.
10. В парламенте 200 депутатов. В процессе заседания произошло 200 потасовок, в каждой из которой участвовали некоторые два депутата. Докажите, что можно объединить в комиссию 67 депутатов, из которых никакие два не выясняли между собой отношения в потасовке.

1.6. Операции над графами

Удаление вершины или ребра, а также переход к подграфу - это операции, с помощью которых можно из имеющегося графа получать другие графы с меньшим числом элементов. Известны также операции, позволяющие, наоборот, получать из имеющихся графов «большие» графы. Такова, например, операция *добавления ребра*: если вершины u, v графа G не смежны, то можно определить граф $G + e$, где $e = uv$. Одной из наиболее важных является *операция объединения*.

Граф H называется *объединением* (или *наложением*) графов F и G , если $V_H = V_F \cup V_G$ и $E_H = E_F \cup E_G$. В этой ситуации пишут $H = F \cup G$. Объединение $F \cup G$ называется *дизъюнктным*, если $V_F \cap V_G = \emptyset$. Аналогично определяются *объединение* и *дизъюнктное объединение* любого множества графов, причем в последнем случае никакие два из объединяемых графов не должны иметь общих вершин.

Пусть $G_i = (V_i, E_i)$ ($i = 1, 2$) - два графа. *Произведением* $G_1 \times G_2 = G$ называется граф, для которого $V_G = V_1 \times V_2$ - декартово произведение множеств вершин исходных графов, а E_G определяется следующим образом: вершины u_1, u_2 и v_1, v_2 смежны в графе G тогда и только тогда, когда или $u_1 = v_1$, а u_2 и v_2 смежны в G_2 , или $u_2 = v_2$, а u_1 и v_1 смежны в G_1 :

$$|G_1 \times G_2| = |G_1| \cdot |G_2|, \quad |E_{G_1 \times G_2}| = |G_1| \cdot |E_{G_2}| + |G_2| \cdot |E_{G_1}|.$$

С помощью *операции произведения* вводится важный класс графов - n -мерные кубы. Определим *n - мерный куб* Q_n рекуррентно: $Q_1 = K_2$, $Q_n = K_2 \times Q_{n-1}$, $n > 1$.

Очевидно, что Q_n - граф порядка 2^n , вершины которого можно представить $(0, 1)$ -векторами длины n таким образом, что две вершины будут смежны тогда и только тогда, когда соответствующие векторы различаются ровно в одной координате. Поскольку каждая вершина n - мерного куба инцидентна n ребрам, то число его ребер равно $n2^{n-1}$.

Еще одна операция - *отождествление* (или - *слияние*) *вершин*. Пусть u, v - две вершины графа G , $H = G - u - v$. К графу H присоединим новую вершину v' , соединив ее ребром с каждой из вершин, входящих в объединение окружений вершин u, v в графе G .

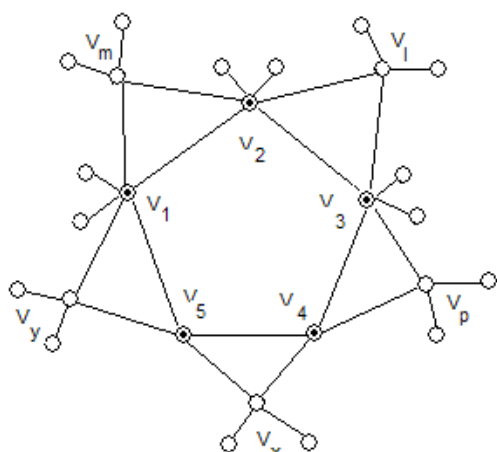


Рис.1.18. Граф G_1 .

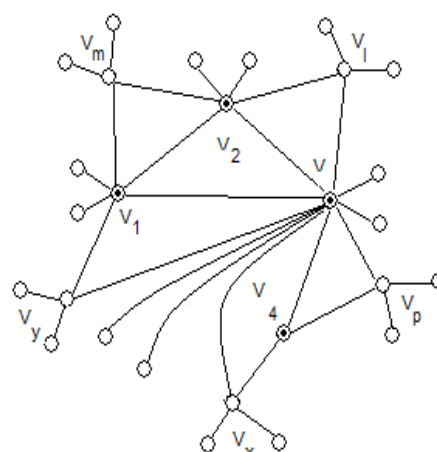


Рис.1.19. Граф G_2 , полученный из G_1 слиянием вершин v_3, v_5 .

Говорят, что построенный граф получается из графа G отождествлением вершин u, v .

Рассматривается также операция стягивания ребра.

Стягивание ребра uv означает отождествление смежных вершин u, v . Граф G называется *стягиваемым к графу H* , если H получается из G в результате некоторой последовательности стягиваний ребер.

Легко видеть, например, что граф Петерсена стягиваем к K_5 и, стало быть, к любому K_n с $n < 5$. Очевидно, что любой непустой связный граф, отличный от K_1 , стягиваем к K_2 . Но уже не любой связный граф стягивается к графу K_3 .

Например, простая цепь P_n не стягивается к K_3 . Естественно возникает параметр $\eta(G)$ - максимум порядков полных графов, к которым стягивается граф G . Параметр $\eta(G)$ называется *числом Хадвигера графа G* . Это число связано с проблемой четырех красок.

В определенном смысле двойственной к операции стягивания ребра является операция *расщепления вершины*.

Пусть v - одна из вершин графа G . Разобьем ее окружение произвольным образом на две части M и N и выполним следующее преобразование графа G : удалим вершину v вместе с инцидентными ей ребрами, добавим новые вершины u, w и соединяющее их ребро uw , вершину u соединим ребром с каждой вершиной из множества M , а вершину w - с каждой вершиной из множества N . Полученный в результате граф обозначим символом \bar{G} . Будем говорить, что \bar{G} получается из графа G расщеплением вершины v .

Задачи и упражнения

1. Для приведенных на рис. 1.20. графов G_1 и G_2 найти $G_1 \cup G_2$, $G_1 \cap G_2$, $G_1 \times G_2$.

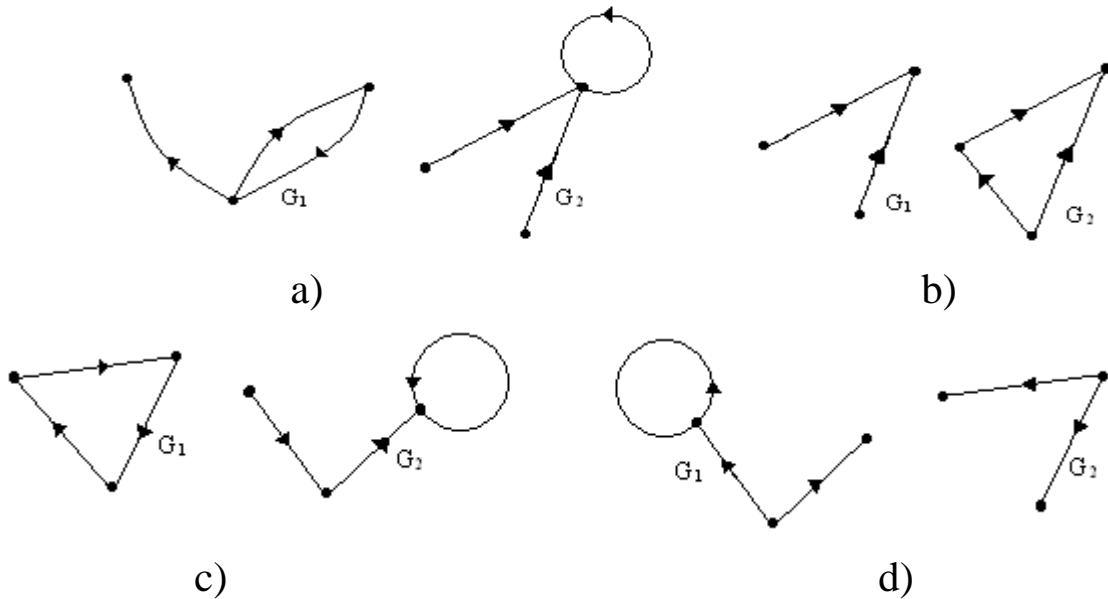


Рис. 1.20.

1.7. Маршруты, цепи, циклы

Чередующаяся последовательность

$$v_1, e_1, v_2, e_2, \dots, e_l, v_{l+1} \quad (1.3)$$

вершин и ребер графа, такая что $e_i == v_i v_{i+1}$ ($i = 1, \dots, l$), называется *маршрутом, соединяющим вершины v_1 и v_{l+1}* (или (v_1, v_{l+1}) - *маршрутом*). Очевидно, что маршрут (1.3) можно задать последовательностью

$$v_1, v_2, \dots, v_{l+1} \quad (1.4)$$

его вершин, а также последовательностью

$$e_1, e_2, \dots, e_l \quad (1.5)$$

ребер. Маршрут называется *цепью*, если все его ребра различны, и *простой цепью*, если все его вершины, кроме, возможно, крайних, различны. Маршрут (1.5) называется *циклическим*, если $v_1 = v_{l+1}$. Циклическая цепь называется *циклом*, а циклическая простая цепь — *простым циклом*. Число l ребер в маршруте (1.4) называется его *длиной*. Простой цикл длины l называется *l -циклом*, 3-цикл часто называют *треугольником*. Длина всякого цикла не менее трех, если речь идет о простом графе, поскольку в таком графе нет петель и кратных ребер. Минимальная из длин циклов графа называется его *обхватом*.

Пусть P - некоторая цепь вида (1.4) в графе G , v_i и v_j - входящие в нее вершины, $i < j$. Очевидно, что часть v_i, v_{i+1}, \dots, v_j цепи

P , начинающаяся в вершине v_i и заканчивающаяся в вершине v_j , сама является цепью графа G . Эта цепь называется (v_i, v_j) - подцепью цепи P .

Для ориентированного графа вводится понятие *ориентированного маршрута* - это последовательность вида (1.5), в которой $e_i = (v_i, v_{i+1})$. Аналогом цепи в этой ситуации служит *путь (ориентированная цепь)*. Вершина v называется *достижимой из вершины u* если существует (u, v) -путь. Поскольку при $u \neq v$ произвольный (u, v) - маршрут, не являющийся простой цепью, превращается в простую (u, v) - цепь после устранения «лишних кусков», то верно

Утверждение 1.1. При $u \neq v$ всякий (u, v) - маршрут содержит простую (u, v) -цепь.

Утверждение 1.2. Всякий цикл содержит простой цикл.

Утверждение 1.3. Объединение двух несовпадающих простых (u, v) -цепей содержит простой цикл.

Утверждение 1.4. Если C и D - два несовпадающих простых цикла, имеющих общее ребро e , то граф $(C \cup D)$ -е также содержит простой цикл.

Путем (или ориентированным маршрутом) ориентированного графа называется последовательность дуг, в которой конечная вершина всякой дуги, отличной от последней, является начальной вершиной следующей.

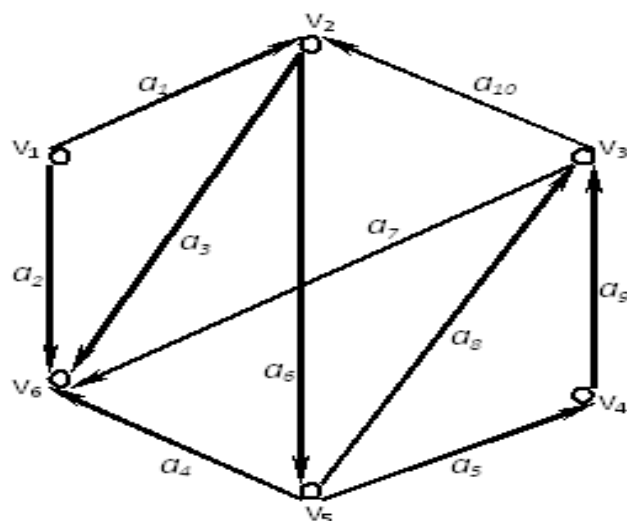


Рис. 1.21. Орграф.

Так, на рис. 1.21 последовательности дуг

$a_6, a_5, a_9, a_8, a_4,$

(1.6)

$$a_1, a_6, a_5, a_9, \quad (1.7)$$

$$a_1, a_6, a_5, a_9, a_{10}, a_6, a_4 \quad (1.8)$$

являются путями.

Дуги $a = (v_i, v_j), v_i \neq v_j$, имеющие общие концевые вершины, называются *смежными*.

Две вершины v_j и v_i называются *смежными*, если какая-нибудь из двух дуг (v_i, v_j) и (v_j, v_i) или обе одновременно присутствуют в графе. Так, например, на рис. 1.21 дуги a_1, a_{10}, a_3, a_6 , как и вершины v_5 и v_3 , являются смежными, в то время как дуги a_1, a_5 или вершины v_1, v_4 не являются смежными.

Ориентированной цепью (или, короче, *орцепью*) называется такой путь, в котором каждая дуга используется не больше одного раза. Так, например, приведенные выше пути (1.6) и (1.7) являются орцепями, а путь (1.8) не является таким, поскольку дуга a_6 в нем используется дважды.

Простой орцепью называется такой путь, в котором каждая вершина используется не более одного раза. Например, путь (1.7) является простой орцепью, а пути (1.6) и (1.8) — нет. Очевидно, что простая орцепь является также орцепью, но обратное утверждение неверно. Например, путь (1.6) является орцепью, но не простой орцепью, путь (1.7) является орцепью и простой орцепью, а путь (1.8) не является ни орцепью, ни простой орцепью.

Маршрут есть неориентированный «двойник» пути, и это понятие рассматривается в тех случаях, когда можно пренебречь направленностью дуг в графе. Таким образом, *маршрут* есть последовательность ребер $\bar{a}_1, \bar{a}_2, \dots, \bar{a}_q$, в которой каждое ребро \bar{a}_i исключением, возможно, первого и последнего ребер, связано с ребрами \bar{a}_{i-1} и \bar{a}_{i+1} своими двумя концевыми вершинами. Последовательности дуг

$$\bar{a}_2, \bar{a}_4, \bar{a}_8, \bar{a}_{10}, \quad (1.9)$$

$$\bar{a}_2, \bar{a}_7, \bar{a}_8, \bar{a}_4, \bar{a}_3 \quad (1.10)$$

$$\bar{a}_{10}, \bar{a}_7, \bar{a}_4, \bar{a}_8, \bar{a}_7, \bar{a}_2 \quad (1.11)$$

в графе, изображенном на рис. 1.22, являются маршрутами; черта над символом дуги означает, что ее ориентацией пренебрегают, т. е. дуга рассматривается как неориентированное ребро.

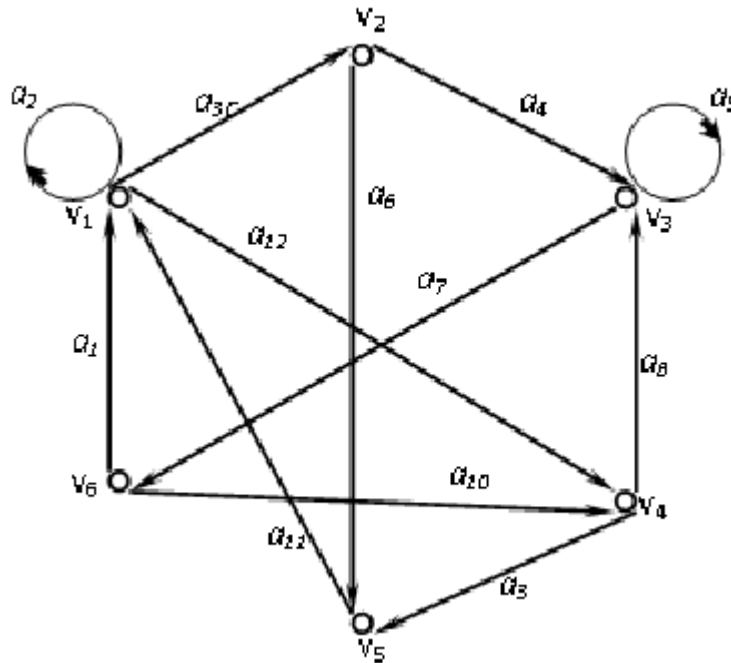


Рис. 1.22. Взвешенный оргграф.

Можно определить цепи и простые цепи. Так, например, маршрут (1.9) есть цепь и простая цепь, маршрут (1.10) — цепь, но не простая цепь, а маршрут (1.11) не является ни цепью, ни простой цепью. Путь или маршрут можно изображать также последовательностью вершин. Путь (1.6) можно представить также последовательностью вершин $v_2, v_5, v_4, v_3, v_8, v_6$.

Иногда дугам графа G дуге (v_i, v_j) ставится в соответствие некоторое число \tilde{p}_{ij} , называемое *весом*, или *длиной*, или *стоимостью (ценой)* дуги. Тогда граф G называется *графом с взвешенными дугами*. Иногда веса (числа v_i) приписываются вершинам v_i графа, и тогда получается граф *с взвешенными вершинами*. Если в графе веса приписаны и дугам, и вершинам, то он называется просто *взвешенным*.

При рассмотрении пути μ , представленного последовательностью дуг $(\bar{a}_1, \bar{a}_2, \dots, \bar{a}_q)$, за его *вес* (или *длину*, или *стоимость*) принимается число $l(\mu)$, равное сумме весов всех дуг, входящих в μ , т.е. $l(\mu) = \sum_{(x_i, x_j) \in \mu} c_{ij}$.

Длиной (или *мощностью*) пути μ называется число его дуг.

Петлей называется дуга, начальная и конечная вершины которой совпадают. На рис. 1.22, например, дуги a_2, a_5 - петли.

Путь a_1, a_2, \dots, a_q называется *замкнутым*, если в нем начальная вершина дуги a_1 совпадает с конечной вершиной дуги a_q .

Так, например, в графе, приведенном на рис. 1.22, последовательности дуг

$$a_3, a_6, a_{11}, \quad (1.12)$$

$$a_{11}, a_3, a_4, a_7, a_1, a_{12}, a_9, \quad (1.13)$$

$$a_3, a_4, a_7, a_{10}, a_9, a_{11} \quad (1.14)$$

являются замкнутыми путями.

Пути (1.12) и (1.13) являются замкнутыми простыми орцепями (*контурами*, или *простыми орциклами*), поскольку в них одна и та же вершина используется только один раз (за исключением начальной и конечной вершин, которые совпадают), а путь (1.13) не является *контуром*, так как вершина x_1 используется в нем дважды. Контур, проходящий через все вершины графа, имеет особое значение и называется *гамильтоновым* контуром. Конечно, не все графы обладают гамильтоновыми контурами. Так, например, контур (1.9) является гамильтоновым контуром графа, приведенного на рис. 1.22, а граф на рис. 1.21 не имеет гамильтоновых контуров, поскольку не существует такой дуги, для которой x_1 была бы конечной вершиной.

Замкнутый маршрут является неориентированным двойником замкнутого пути. Таким образом, замкнутый маршрут является маршрутом v_1, v_2, \dots, v_q , в котором совпадают начальная и конечная вершины, т. е. в котором $x_1 = x_q$.

Задачи и упражнения

1. Докажите, что если в графе больше 5 вершин, либо сам граф, либо его дополнение содержат цикл длины 3.
2. Приведите пример графа на 5 вершинах, для которого утверждение предыдущего пункта неверно.
3. Сколько вершин должно быть в графе, чтобы либо он сам, либо его дополнение обязательно содержали хоть какой-нибудь цикл?
4. Степень каждой из 20 вершин графа G не меньше 14. Докажите, что найдется 4 вершины, попарно соединенные между собой. (Научно: в G есть подграф, изоморфный K_4 .)
5. В простом связном графе n вершин, $n > 7$, и степени всех вершин равны 3. Докажите, что в этом графе есть простой путь длины 6.
6. Доказать, что минимальное количество ребер в простом связном графе с n вершинами равно $n - 1$.
7. Доказать, что каждый связный граф с n вершинами и не менее, чем с n ребрами, имеет цикл, а если ребер ровно n , то этот цикл единственный.
8. В графе все вершины имеют степень 3. Докажите, что в нем есть цикл.
9. Пусть каждая вершина графа G без петель имеет степень не менее 3. Доказать, что граф имеет цикл четной длины. (Указание: рассмотреть максимальный путь.)

1.8. Связность и компоненты

Две вершины $v_i, v_j \in V$ графа $G=(V, E)$ называются связанными, если в нём существует (v_i, v_j) -маршрут. Вершина связана сама с собой.

Учитывая утверждение 1.1, в этом определении можно заменить маршрут цепью или простой цепью.

Граф $G=(V, E)$ называется *связным*, если каждая пара различных вершин $v_i, v_j \in V$ связана. Пусть V - множество вершин графа G . Говорят, что подмножества V_1, V_2, \dots, V_n образуют разбиение множества V , если

$$1) V_i \cap V_j = \emptyset, \forall i \neq j, i, j = 1, 2, \dots, q;$$

$$2) \bigcup_{r=1}^q V_r = V.$$

Если разбиение V_1, V_2, \dots, V_n удовлетворяет условию $|V_r|=h, r=1, 2, \dots, q$, то оно называется h -разбиением множества V . Разбиения множества V называют упорядоченными, если равенство разбиений $V = \bigcup_{r=1}^q V_r$ и $V = \bigcup_{r=1}^q V'_r$ означают, что $V_r = V'_r, \forall r=1, 2, \dots, q$.

Если граф $G=(V, E)$ – не является связным, то его множество V вершин можно разбить на такие подмножества V_1, V_2, \dots, V_p , что вершинно - порождённые подграфы $\langle V_i \rangle, i=1, 2, \dots, p$ связны, и никакая вершина $v_k \in V_i$ не связана ни с какой вершиной $v_m \in V_j, i \neq j$. При этом подграфы $\langle V_i \rangle, i=1, 2, \dots, p$ называются компонентами графа $G=(V, E)$. Из определения следует, что компонента связности графа G является его максимальным связным подграфом. Например, на рис. 1.23 граф G - не связный и имеет 4 компоненты.

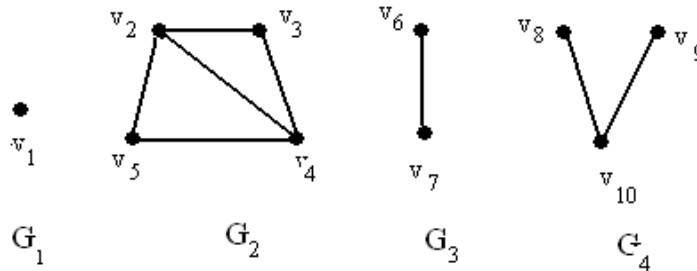


Рис.1.23. Граф G и его компоненты $G_i, i=1, 2, 3, 4$.

Теорема 1.3. В связном графе $G=(V, E)$ любые две цепи максимальной длины имеют общую вершину.

Доказательство. Пусть P_1, P_2 - две цепи максимальной длины в графе G : $P_1 = (v_0, v_1, v_2, \dots, v_j, \dots, v_k)$, $P_2 = (v'_0, v'_1, v'_2, \dots, v'_j, \dots, v'_k)$. Подцепи P_1, P_2 -цепей обозначим через: $P_{1,1} = (v_0, v_i)$ - цепь, $P_{1,2} = (v_i, v_k)$ -цепь, $P_{2,1} = (v'_0, v'_j)$ - цепь, $P_{2,2} = (v'_j, v'_k)$ -цепь, причём $t_1 = |P_{1,1}|$, $t_2 = |P_{1,2}|$, $t'_1 = |P_{2,1}|$, $t'_2 = |P_{2,2}|$ - длины этих цепей соответственно (рис. 1.24).

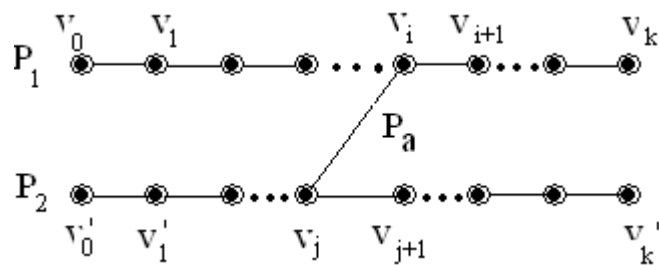


Рис. 1.24. К доказательству теоремы 1.3.

Теорему 1.3 докажем от противного. Предположим, что цепи P_1, P_2 не имеют общую вершину, т.е. $P_1 \cap P_2 = \emptyset$. Поскольку граф G связный, то для некоторых $i, 0 \leq i \leq k$ существует (v_i, v_j) -цепь, соединяющая вершины $v_i \in P_1, v_j \in P_2$. Обозначим её через P_α ($\alpha > 1$), причём $t_\alpha = |P_\alpha|$ - длина цепи P_α . Не ограничивая общности, предположим, что выполняются неравенства: $t_1 \geq t'_1; t_2 \geq t'_2$. Тогда в графе G существует цепь $P_{1,1} \cup P_\alpha \cup P_{2,1}$, длина которой больше длины максимальной цепи: $|P_{1,1} \cup P_\alpha \cup P_{2,1}| = t_1 + t_\alpha + t'_2 > t'_1 + t'_2 = |P_2|$. Получили противоречие. Теорема 1.3 доказана.

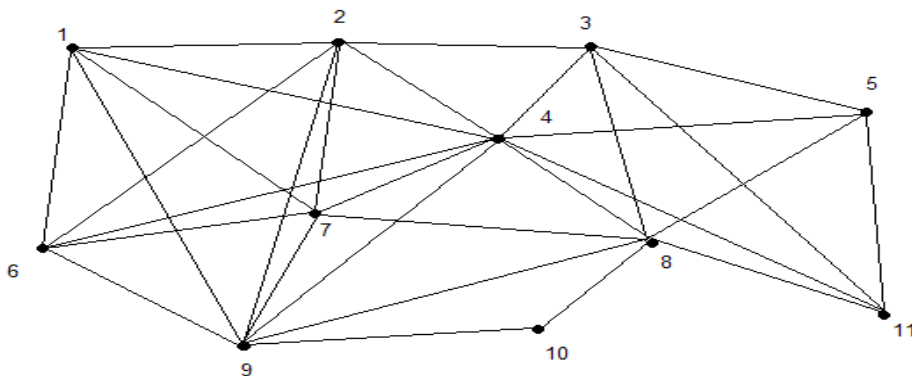


Рис.1.25. Связный граф.

Утверждение 1.5. Для связности графа необходимо и достаточно, чтобы в нем для какой-либо фиксированной вершины и каждой другой вершины v существовал (u, v) -маршрут.

Всякий максимальный связный подграф графа G называется *связной компонентой* (или просто *компонентой*) графа G . Слово «максимальный» означает максимальный относительно включения, т.е. не содержащийся в связном подграфе с большим числом элементов. Множество вершин связной компоненты называется *областью связности* графа.

Утверждение 1.6. Каждый граф представляется в виде дизъюнктного объединения своих связных компонент. Разложение графа на связные компоненты определено однозначно.

Утверждение 1.7. Для любого графа либо он сам, либо его дополнение является связным.

Лемма 1.3. Пусть G — связный граф, $e \in EG$. Тогда:

1) если ребро e принадлежит какому-либо циклу графа G , то граф $G - e$ — e связан;

2) если ребро e не входит ни в какой цикл, то граф $G - e$ имеет ровно две компоненты.

Ниже число ребер и число компонент графа G обозначаются через $m(G)$ и $k(G)$ соответственно.

Лемма 1.4. Если $k(G)=k$ для n -вершинного графа G , то
$$n - k \leq m(G) \leq (n - k)(n - k + 1) / 2, \quad (1.15)$$

причем обе эти оценки для $m(G)$ достижимы.

Следствие 1.1. При фиксированных n и $k \leq n$ среди графов G порядка n с $k(G)=k$ существует только один граф, а именно, $G = O_{k-1} \cup K_{n-k+1}$, с максимальным числом ребер.

1.9. Сильная связность

Ориентированный граф называется *сильно связным* или *сильным*, если для любых двух различных вершин v_i и v_j

существует по крайней мере один путь, соединяющий v_i с v_j . Это определение означает также, что любые две вершины такого графа *взаимно достижимы*.

Ориентированный граф называется *односторонне связным* или *односторонним*, если для любых двух различных вершин v_i и v_j существует по крайней мере один путь из v_i в v_j или из v_j в v_i (или оба одновременно).

Ориентированный граф называют *слабо связным* или *слабым*, для любых двух различных вершин графа существует, по крайней мере, один маршрут, соединяющий их.

Если для некоторой пары вершин орграфа не существует маршрута, соединяющего их, то такой орграф называется *несвязным*. Пусть дано некоторой свойство P , которым могут обладать графы.

Максимальным подграфом графа G относительно свойства P называется порожденный подграф X_S графа G , обладающий этим свойством и такой, что не существует другого порожденного подграфа $\langle X'_S \rangle$, у которого $X_S \supset X'_S$, и который также обладает свойством P .

Так, например, если в качестве свойства P взята сильная связность, то максимальным сильным подграфом графа G является сильный подграф, который не содержится в любом другом сильном подграфе. Такой подграф называется *сильной компонентой* графа G .

Аналогично, *односторонняя компонента* представляет собой односторонний максимальный подграф, а *слабая компонента* — максимальный слабый подграф.

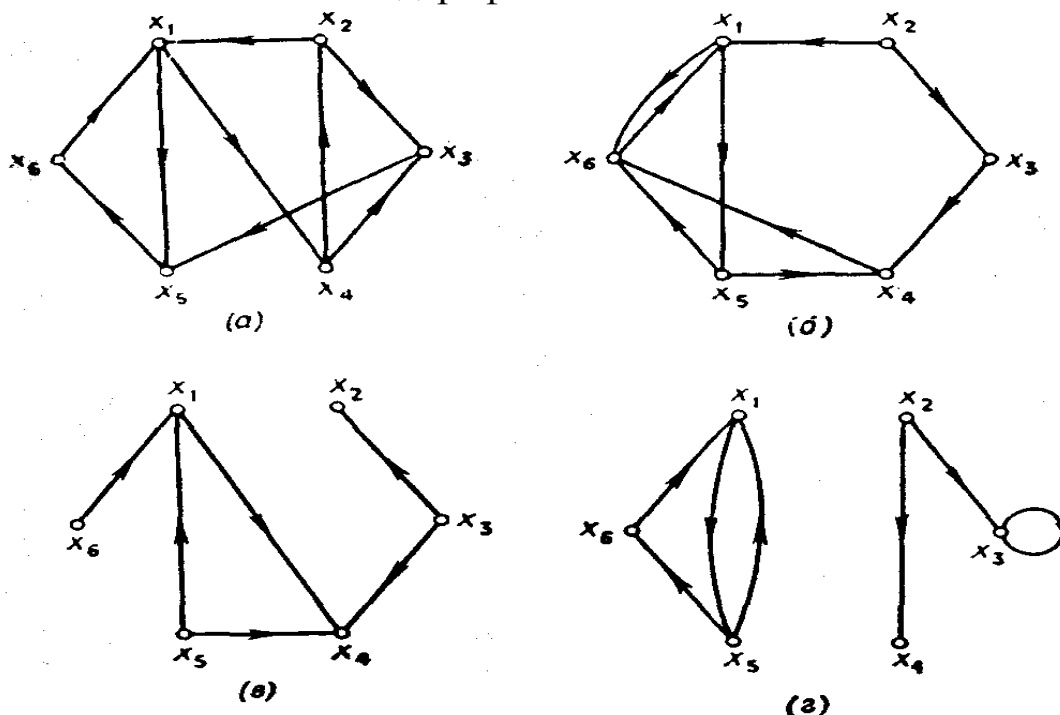


Рис. 1.26. Граф: (а) сильно связный, (б) односторонне связный (в) слабо связный, (г) несвязный.

Например, в графе G , приведенном на рис. 1.26(б), подграф $\langle \{x_1, x_4, x_5, x_6\} \rangle$ является сильной компонентой графа G . С другой стороны, подграфы $\langle \{x_1, x_6\} \rangle$ и $\langle \{x_1, x_5, x_6\} \rangle$ не являются сильными компонентами (хотя и являются сильными подграфами), поскольку они содержатся в графе $\langle \{x_1, x_4, x_5, x_6\} \rangle$ и, следовательно, не максимальные. В графе, показанном на рис. 1.26(в), подграф $\langle \{x_1, x_4, x_5\} \rangle$

является односторонней компонентой. В графе, приведенном на рис. 1.26(г), оба подграфа $\langle\{x_1, x_5, x_6\}\rangle$ и $\langle\{x_1, x_5, x_6\}\rangle$ являются слабыми компонентами, и у этого графа только две такие компоненты.

Из определений следует, что односторонние компоненты графа могут иметь общие вершины. Сильная компонента должна содержаться, по крайней мере, в одной односторонней компоненте, а односторонняя компонента содержится в некоторой слабой компоненте данного графа G .

Задачи и упражнения

1. Связность графа и связность БО. Пусть для вершин связного графа G на n вершинах выполнено следующее свойство: если a соединена с b , а b соединена с c , то a соединена с c . Найдите число ребер графа.
2. В 15-этажном доме имеется лифт с двумя кнопками: "+7" и "-9". Можно ли проехать с этажа на любой?
3. Докажите, что граф с n вершинами, степень каждой из которых не менее $(n-1)/2$, - связан.
4. Граф со 100 вершинами имеет 98 вершин степени 30, и по одной вершине степеней 25 и 15. Обязательно ли он будет связан? Докажите, что вершины степеней 25 и 15 лежат в одной компоненте связности.
5. В связном графе степени всех вершин четные. Докажите, что граф останется связным и после удаления любого из ребер.
6. В графе никакие два цикла не имеют общих вершин. Может ли в таком графе быть 3 цикла, 16 вершин и 17 ребер?
7. В графе 8 вершин и 22 ребра. Верно ли, что этот граф- связный?
8. Обобщая, а) Сколько ребер может быть у связного графа на n вершинах? б) Скольких ребер заведомо достаточно, чтобы граф был связан?
9. а) Каждое из ребер полного графа с 6 вершинами покрашено в один из двух цветов. Докажите, что есть три вершины, все ребра между которыми - одного цвета.
10. Составить программу которая проверяет граф на связность. Если он несвязный, то найти число компонент связности.

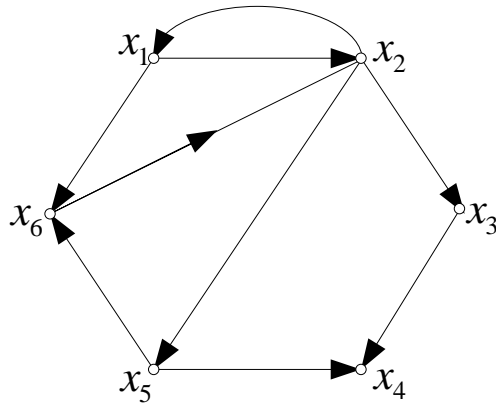


Рис. 1.27. Ориентированный граф G .

1.10. Матрицы достижимостей

Матрица достижимостей $R = [r_{ij}]$ определяется следующим образом:

$$r_{ij} = \begin{cases} 1, & \text{если вершина } x_j \text{ достижима из } x_i, \\ 0, & \text{в противном случае.} \end{cases}$$

Множество вершин $R(x_i)$ графа G , достижимых из заданной вершины x_i , состоит из таких элементов x_j , для которых r_{ij} в матрице достижимостей равно 1. Все диагональные элементы в матрице R равны 1, поскольку каждая вершина достижима из себя самой с помощью пути длины 0.

Поскольку $\Gamma(x_i)$ является множеством таких вершин x_j , которые достижимы из x_i с использованием путей длины 1, то $\Gamma(x_i)$ является множеством вершин, для которых в графе существуют дуги (x_i, x_j) . Соответственно множество $\Gamma(\Gamma(x_i)) = \Gamma^2(x_i)$ состоит из вершин, достижимых из x_i с использованием путей длины 2. Аналогично $\Gamma^p(x_i)$ является множеством вершин, которые достижимы из x_i с помощью путей длины p . Так как любая вершина графа G , которая достижима из x_i должна быть достижима с использованием пути (или путей) длины 0, 1, 2, ..., p (число p меньше числа вершин в графе), то множество вершин достижимых из x_i , можно представить в виде

$$R(x_i) = \{x_i\} \cup \Gamma(x_i) \cup \Gamma^2(x_i) \cup \dots \cup \Gamma^p(x_i). \quad (1.16)$$

Таким образом, множество $R(x_i)$ может быть получено последовательным выполнением (слева направо) операций объединения в соотношении (1.16), до тех пор, пока текущее множество не перестанет увеличиваться по мощности при очередной операции

объединения. С этого момента последующие операции не будут давать новых членов множеству и, таким образом, будет образовано достижимое множество $R(x_i)$.

Матрицу достижимостей можно построить следующим образом. Находим достижимые множества $R(x_i)$ для всех вершин $x_i \in X$ способом, приведенным выше. Положим $r_{ij}=1$, если $x_j \in R(x_i)$, и $r_{ij}=0$ в противном случае. Полученная таким образом матрица R является матрицей достижимостей.

Матрица контрдостижимостей $Q = [q_{ij}]$ определяется следующим образом:

$$q_{ij} = \begin{cases} 1, & \text{если вершина } x_i \text{ достижима из вершины } x_j, \\ 0, & \text{в противном случае.} \end{cases}$$

Контрдостижимым множеством $Q(x_i)$ графа G является множество таких вершин, что из любой вершины этого множества достижимы вершины x_i . Аналогично построению достижимого множества $R(x_i)$ на основе соотношения (1.16) можно сформировать множество $Q(x_i)$, используя следующее выражение:

$$Q(x_i) = \{x_i\} \cup \Gamma^{-1}(x_i) \cup \Gamma^{-2}(x_i) \cup \dots \cup \Gamma^{-p}(x_i), \quad (1.17)$$

где $\Gamma^{-2}(x_i) = \Gamma^{-1}(\Gamma^{-1}(x_i))$ и т. д. Операции выполняются слева направо до тех пор, пока очередная операция объединения не перестанет изменять текущее множество $Q(x_i)$.

Следует заметить, что столбец x_i матрицы Q совпадет со строкой x_i матрицы R , т. е. $Q = R^T$, где R^T – матрица, транспонированная к матрице достижимостей R .

Пример 1.8. Найти матрицы достижимостей и обратных достижимостей для графа G , приведенного на рис. 1.28.

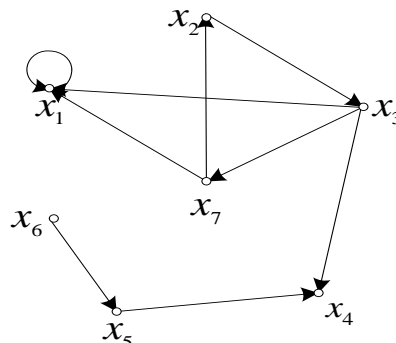


Рис. 1.28. Ориентированный граф G .

Матрица смежности данного графа имеет вид

$$A = \begin{bmatrix} & x1 & x2 & x3 & x4 & x5 & x6 & x7 \\ x1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x2 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ x3 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ x4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ x5 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ x6 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ x7 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Множества достижимостей найдем с помощью (1.16):

$$R(x_1) = \{x_1\} \cup \{x_1\} = \{x_1\}$$

$$R(x_2) = \{x_2\} \cup \{x_3\} \cup \{x_1, x_4, x_7\} \cup \{x_2, x_4\} \cup \{x_1\} = \{x_1, x_2, x_3, x_4, x_7\}$$

$$R(x_3) = \{x_3\} \cup \{x_1, x_4, x_7\} \cup \{x_1, x_2\} \cup \{x_1, x_3\} = \{x_1, x_2, x_3, x_4, x_7\}$$

$$R(x_4) = \{x_4\} = \{x_4\}$$

$$R(x_5) = \{x_5\} \cup \{x_4\} = \{x_4, x_5\}$$

$$R(x_6) = \{x_6\} \cup \{x_5\} \cup \{x_4\} = \{x_4, x_5, x_6\}$$

$$R(x_7) = \{x_7\} \cup \{x_1, x_2\} \cup \{x_1, x_3\} \cup \{x_1, x_4, x_7\} \cup \{x_1, x_2\} = \{x_1, x_2, x_3, x_4, x_7\}.$$

Следовательно, матрицы достижимостей и обратных достижимостей имеют вид

$$Q = R^T = \begin{bmatrix} & x1 & x2 & x3 & x4 & x5 & x6 & x7 \\ x1 & 1 & 1 & 1 & 0 & 0 & 0 & 1 \\ x2 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ x3 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \\ x4 & 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ x5 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \\ x6 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ x7 & 0 & 1 & 1 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Нахождение матриц R и Q с вычислительной точки зрения является довольно простой задачей, поскольку объединение множеств в соответствии с выражениями (1.16) и (1.17) и сравнение текущих множеств после каждого объединения, проводимое для выяснения необходимости продолжения процесса построения соответствующих множеств - все это можно осуществить в математических пакетах и системах программирования.

Так как $R(x_i)$ является множеством вершин, достижимых из x_i , а $Q(x_j)$ - множеством вершин, из которых можно достигнуть x_j , то $R(x_i) \cap Q(x_j)$ - множество таких вершин, каждая из которых принадлежит, по крайней мере, одному пути, идущему от x_i к x_j . Эти вершины называются существенными относительно двух концевых

вершин x_i и x_j . Все остальные вершины $x_k \notin R(x_i) \cap Q(x_j)$ называются несущественными, поскольку их удаление не влияет на пути от x_i к x_j .

Матрицы достижимостей и обратных достижимостей, определенные выше, являются полными в том смысле, что на длины путей от x_i к x_j не накладывались никакие ограничения. С другой стороны, можно определить матрицы ограниченных достижимостей и контрдостижимостей – надо потребовать, чтобы длины путей не превышали некоторого заданного числа. Эти ограниченные матрицы тоже могут быть построены с помощью соотношений (1.16) и (1.17) – надо действовать точно так, как раньше, при нахождении неограниченных матриц, но только теперь p будет верхней границей длины допустимых путей.

1.11. Нахождение сильных компонент и построение конденсации

Сильная компонента (СК) графа G является максимальным сильно связным подграфом графа G . Поскольку в сильно связном графе произвольная вершина x_j достижима из любой другой вершины x_i , то в ориентированном графе существует одна и только одна СК, содержащая данную вершину x_i . В самом деле, если бы вершина x_i принадлежала двум или большему числу сильных компонент, то существовал бы путь из любой вершины одной СК в произвольную вершину другой СК и, следовательно, объединение этих сильных компонент было бы сильно связным графом, что противоречит определению СК.

Если вершина x_i одновременно является начальной и конечной вершиной пути, то множество вершин, существенных относительно этих двух идентичных концов (т.е. множество вершин некоторого цикла, содержащего x_i), совпадает с пересечением $R(x_i) \cap Q(x_i)$. Поскольку все эти существенные вершины достижимы из x_i и, кроме того, из каждой такой вершины достижима вершина x_i , то все они взаимно достижимы. Более того, если нет другой вершины, существенной относительно концов x_i и x_i , то множество $R(x_i) \cap Q(x_i)$, которое может быть построено с использованием соотношений (1.16) и (1.17) однозначно определяет СК графа G , содержащую вершину x_i .

Если эти вершины удалить из графа $G=(X, \Gamma)$, то в оставшемся порожденном подграфе $G' = \langle X - R(x_i) \cap Q(x_i) \rangle$ можно таким же

способом выделить новую СК, содержащую $x_j \in X - R(x_i) \cap Q(x_i)$. Эту процедуру можно повторять до тех пор, пока все вершины графа G не будут сгруппированы в соответствующие СК. После завершения этой процедуры граф G будет разбит на свои сильные компоненты.

Граф $G^*=(X^*, \Gamma^*)$ называют конденсацией графа G и определяют следующим образом: каждая его вершина представляет множество вершин некоторой сильной компоненты графа G , дуга (x_i^*, x_j^*) существует в G^* тогда и только тогда, когда в G существует дуга (x_i, x_j) , такая, что x_i принадлежит компоненте, соответствующей вершине x_i^* , а x_j – компоненте, соответствующей вершине x_j^* .

Конденсация G^* не содержит циклов, поскольку наличие цикла означает, что любые вершины этого цикла взаимно достижимы, а поэтому совокупность всех вершин цикла принадлежит некоторой СК в G^* и, следовательно, содержится в СК графа G , что противоречит определению конденсации, в силу которого вершины из G^* соответствуют СК в G .

Пример 1.9. Для графа G , приведенного на рис. 1.29 найти сильные компоненты и построить конденсацию G^* .

Из соотношений (1.16) и (1.17) получаем:

$R(x_1)=\{x_1, x_2, x_3, x_4\}$, $R(x_2)=\{x_1, x_2, x_3, x_4\}$, $R(x_3)=\{x_1, x_2, x_3, x_4\}$,
 $R(x_4)=\{x_1, x_2, x_3, x_4\}$, $R(x_5)=\{x_1, x_2, x_3, x_4, x_5\}$, $R(x_6)=\{x_1, x_2, x_3, x_4, x_6,$
 $x_7, x_8, x_9, x_{10}\}$, $R(x_7)=\{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9, x_{10}\}$, $R(x_8)=\{x_8\}$,
 $R(x_9)=\{x_1, x_2, x_3, x_4, x_6, x_7, x_8, x_9, x_{10}\}$, $R(x_{10})=\{x_8, x_{10}\}$ и $Q(x_1)=\{x_1, x_2,$
 $x_3, x_4\}$, $Q(x_2)=\{x_1, x_2, x_3, x_4\}$, $Q(x_3)=\{x_1, x_2, x_3, x_4\}$, $Q(x_4)=\{x_1, x_2, x_3,$
 $x_4\}$, $Q(x_5)=\{x_5\}$, $Q(x_6)=\{x_6, x_7, x_9\}$, $Q(x_7)=\{x_6, x_7, x_9\}$, $Q(x_8)=\{x_8\}$,
 $Q(x_9)=\{x_6, x_7, x_9\}$, $Q(x_{10})=\{x_{10}\}$.

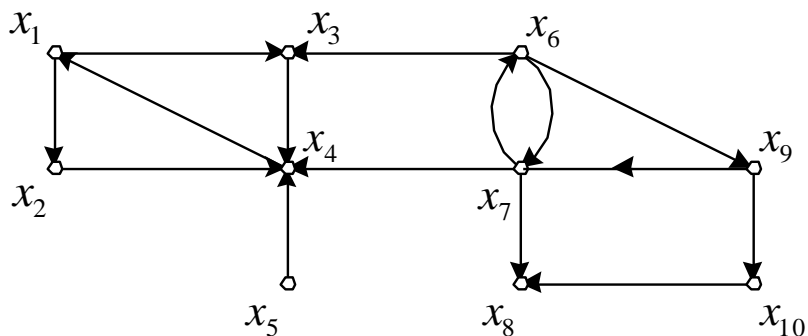


Рис. 1.29. Ориентированный граф G .

Найдем СК в G , содержащую вершину x_1 .

$$\langle R(x_1) \cap Q(x_1) \rangle = \langle \{x_1, x_2, x_3, x_4\} \rangle.$$

Аналогично, СК, содержащая вершину x_6 , есть порожденный подграф $\langle \{x_6, x_7, x_9\} \rangle$, СК содержащая x_5 - подграф $\langle \{x_5\} \rangle$, СК, содержащая x_8 - подграф $\langle \{x_8\} \rangle$ и СК, содержащая x_{10} - подграф $\langle \{x_{10}\} \rangle$. Следует отметить, что три последних СК состоят из единственной вершины графа G . Конденсация G^* приведена на рис. 1.30.

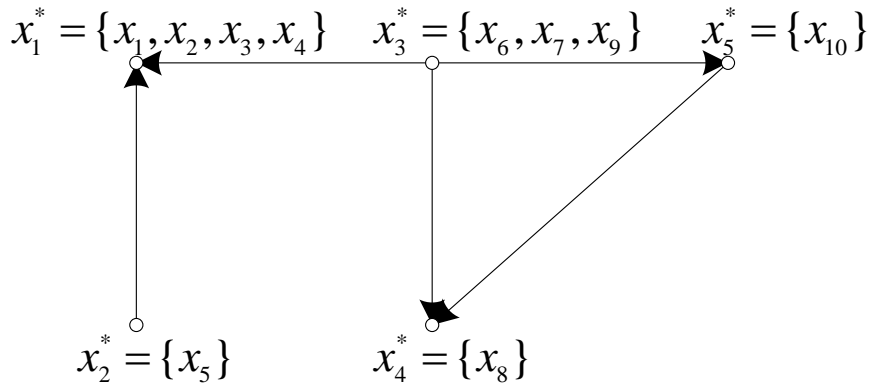


Рис. 1.30. Граф G^* - конденсация графа G .

Процедуру, описанную выше и связанную с нахождением СК графа, можно сделать более удобной, если непосредственно использовать матрицы достижимостей R и контрдостижимостей Q . Выполним поэлементное умножение матриц R и Q ($R \otimes Q$). Строка x_i матрицы $R \otimes Q$ содержит единицы только в тех столбцах x_j , для которых выполняется условие: вершины x_i и x_j взаимно достижимы; в других местах строки x_i стоят нули. Таким образом, две вершины находятся в одной и той же СК тогда и только тогда, когда соответствующие им строки (или столбцы) в матрице $R \otimes Q$ идентичны. Вершины, которым соответствуют строки, содержащие единицу в столбце x_j образуют множество вершин СК, содержащей x_j :

$$R \otimes Q = \begin{bmatrix} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 & x_{10} \\ x_1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_2 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_3 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_4 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ x_5 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ x_6 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ x_7 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ x_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ x_9 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \\ x_{10} & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Отсюда следует, что матрицу $R \otimes Q$ можно преобразовать путем сортировки строк и столбцов в блочно-диагональную; каждая из диагональных подматриц этой матрицы соответствует СК графа G и содержит только единичные элементы; все остальные элементы блочно-диагональной матрицы равны нулю. Для приведенного ранее примера матрица $R \otimes Q$, преобразованная соответствующим образом, имеет вид

$$R \otimes Q = \begin{bmatrix} & x1 & x2 & x3 & x4 & x5 & x6 & x7 & x9 & x8 & x10 \\ x1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & & \\ x2 & 1 & 1 & 1 & 1 & & & & & & \\ x3 & 1 & 1 & 1 & 1 & & & & & & \\ x4 & 1 & 1 & 1 & 1 & & & & & & \\ x5 & 0 & 1 & 0 & 0 & 0 & & & & & \\ x6 & 0 & 0 & & & & & 1 & 1 & 1 & 0 \\ x7 & & & & & & & 1 & 1 & 1 & \\ x9 & & & 0 & & & & 1 & 1 & 1 & \\ x8 & 0 & 0 & 0 & 0 & 0 & & & & 1 & 0 \\ x10 & 0 & 0 & 0 & 0 & 0 & & & & 0 & 1 \end{bmatrix}$$

1.12. Базы и антибазы

База B графа есть множество вершин, из которого достижима любая вершина графа и которое является минимальным в том смысле, что не существует собственного подмножества в B , обладающего таким свойством достижимости. Если мы обозначим через $R(B)$ множество вершин, достижимых из вершин множества B , т.е. если $R(B) = \bigcup_{x_i \in B} R(x_i)$, то B является базой тогда и только то-

гда, когда $R(B) = X$ и $\forall B' \subset B, R(B') \neq X$.

Второе условие $\forall B' \subset B, R(B') \neq X$ эквивалентно утверждению: $x_j \notin R(x_i)$ для любых двух различных $x_i, x_j \in B$, т.е. вершина из B не достижима из любой другой вершины B .

Итак, базой является такое множество B вершин графа G , которое удовлетворяет следующим двум условиям: каждая вершина графа G достижима хотя бы из одной вершины множества B , и в B нет вершины, которая достижима из другой вершины множества B . Из этих двух условий получаются следующие утверждения: в

множестве B нет двух вершин, которые принадлежат одной и той же СК графа G и в любом графе без циклов существует единственная база; состоящая из всех вершин графа с полустепенями захода, равными 0.

Таким образом, в силу этих утверждений база B^* конденсации G^* графа G состоит из таких вершин графа G^* , полустепени захода которых равны нулю. Следовательно, базы графа G можно строить так: из каждой СК графа G , соответствующей вершине базы B^* конденсации G^* надо взять по одной вершине, т.е. если $B^* = \{S_1, S_2, \dots, S_m\}$, где m - число вершин-множеств S_j в базе B^* графа G^* , то базой B является произвольное множество $\{x_{i_1}, x_{i_2}, \dots, x_{i_m}\}$, где $x_{i_j} \in S_j$.

Антибаза \bar{B} есть множество вершин графа G , таких, что

$$\left. \begin{aligned} Q(\bar{B}) &= \bigcup_{x_i \in \bar{B}} Q(x_i) = X \\ \forall B' \subseteq \bar{B}, Q(B') &\neq X \end{aligned} \right\}$$

т.е. \bar{B} есть такое минимально возможное множество вершин, что какова бы ни была вершина графа G , из нее достижима некоторая вершина в \bar{B} . Свойства антибаз аналогичны свойствам баз, надо только прямые понятия заменить на двойственные.

Таким образом, антибаза конденсации G^* есть множество вершин в G^* , полустепени исхода которых равны 0, и антибазы самого графа G строятся из антибазы графа G^* путем выбора по одной вершине в каждой вершине-множестве антибазы \bar{B}^* - подобно тому, как это делалось раньше для баз.

Пример.1.10. Для графа G , приведенного на рис. 1.29, конденсация G^* показана на рис. 1.30. Базой графа G^* является множество $\{x_2^*, x_3^*\}$, поскольку x_2^* и x_3^* - единственные вершины в G^* с полустепенями захода, равными 0. Базами графа G являются $\{x_5, x_6\}$, $\{x_5, x_7\}$ и $\{x_5, x_9\}$.

В примере с графом G , изображенным на рис. 1.29, конденсация G^* (рис. 1.30) содержит вершины x_1^* и x_4^* с полустепенью исхода, равной 0. Таким образом, антибазой графа G^* является множество $\{x_1^*, x_4^*\}$, а антибазами графа G являются множества $\{x_8, x_1\}$, $\{x_8, x_2\}$, $\{x_8, x_3\}$ и $\{x_8, x_4\}$.

1.13. Применение к исследованию структуры организаций

Если граф G представляет структуру руководства или влияний некоторой организации, то члены каждой сильной компоненты графа G имеют равную власть или равное влияние друг на друга, как это может быть, например, для случая комитета. Базу графа G можно интерпретировать как «коалицию», включающую наименьшее число лиц, обладающих властью над каждым членом организации.

Пусть на множестве вершин, представляющих членов той же самой организации, построен новый граф G' , отображающий каналы связи, так что каждая дуга (x_i, x_j) означает, что x_i может связываться с x_j . Граф G' , конечно, каким-то образом связан с графом G , но совсем не очевидным образом. Наименьшее число лиц, которые знают или могут получить все сведения об организации, образует одну из антибаз графа G' . Можно утверждать, что эффективная для управления этой организацией коалиция будет множеством лиц H , определяемым следующим соотношением: $H = B(G) \cup \bar{B}(G')$, где $B(G)$ и $B(G')$ - одна из баз графа G и одна из антибаз графа G' , выбранные так, чтобы $|H|$ (число людей в H) было минимальным.

Приведенное выше описание организации с использованием теории графов является упрощенным. Один из недостатков состоит в том, что нежелательно, чтобы лицо, не входящее в B , имело бы власть над лицом из B .

Следовательно, можно определить сильную базу как такое множество вершин $B_p \subseteq X$, что

$$R(B_p) = X, \quad Q(B_p) = B_p$$

и

$$\forall B' \subset B_p, \quad R(B') \neq X.$$

Вторая часть условия выражает тот факт, что только лица из B_p могут иметь власть над другими лицами, также принадлежащими B_p , и может быть заменено эквивалентным условием $R(X - B_p) \cap B_p = \emptyset$. Это условие означает, что если вершина из СК графа G входит в B_p , то и каждая вершина из той же самой СК должна входить в B_p . Поскольку база в G^* есть множество таких вершин, полустепени захода которых равны 0, т. е. ни одна из этих вершин не достижима ни из какой другой вершины графа, то силь-

ная база в G есть объединение множеств вершин базы графа G^* , т.е. $B_p = \bigcup_{S_j \in B^*} S_j$.

Для графа, приведенного в примере на рис. 1.29, сильная база G есть $\{x_5, x_6, x_7, x_9\}$. Можно отметить, что, если этот граф представляет организацию, то $\{x_6, x_7, x_9\}$ можно рассматривать как совет директоров, обладающий властью над всеми множествами лиц x_1^* , x_4^* и x_5^* , в то время как $\{x_5\}$ можно рассматривать сотрудника, имеющего власть над множеством лиц x_1^* .

2. ДЕРЕВЬЯ

Граф $G = (V, E)$ называется деревом, если он связный и не содержит циклов. Вершины степени 1 в дереве называют его листьями. Подграф $G_1 = (V_1, E_1)$ графа $G = (V, E)$ называется остовным деревом, если G_1 - дерево и $V_1 = V$.

Лемма 2.1. Любой (конечный) связный граф $G = (V, E)$ содержит хотя бы одно остовное дерево $G_1 = (V, E_1)$.

Доказательство. Если в G нет циклов, то положим $G_1 = G$. Если в G есть циклы, то удалим из G какое-нибудь ребро, входящее в цикл. Получится некоторый подграф G . По лемме 1.1 G - связный граф. Если в G нет циклов, то положим $G_1 = G$, иначе продолжим этот процесс. Процесс должен завершиться, так как E - конечное множество.

Лемма 2.2. Пусть в дереве G имеется p вершин и q ребер. Тогда $q = p - 1$.

Доказательство. Так как G - связный граф G не содержит циклов, то $p - q = 1$ по лемме 1.4. Отсюда $q = p - 1$.

Понятие дерева можно определить различными способами, что вытекает из следующей теоремы.

Теорема 2.1. Пусть $G = (V, E)$ - неориентированный граф без петель и кратных ребер, $|V| = p, |E| = q$. Тогда следующие условия эквивалентны:

- 1) G - дерево;
- 2) G - без циклов и $q = p - 1$;
- 3) G - связный и $q = p - 1$;
- 4) G - связный, но при удалении любого ребра становится не-связным;

5) G - без циклов, но при добавлении любого нового ребра на тех же вершинах появляется цикл.

Доказательство. Докажем следующие переходы 1) \rightarrow 2) \rightarrow 3) \rightarrow 4) \rightarrow 5) \rightarrow 1), откуда будет следовать, что из любого условия вытекает любое другое.

Переход 1) \rightarrow 2) следует из леммы 2.2. Пусть теперь выполняется 2). Тогда по лемме 1.1 в G число связных компонент равно $q=p-1$, то есть G - связный граф. Отсюда следует переход 2) \rightarrow 3). Переход 3) \rightarrow 4) вытекает из следствия к лемме 2.2. Пусть выполняется 4). Тогда если бы в G был цикл, то при удалении любого ребра из этого цикла G остался бы связным, согласно лемме 1.3. Это противоречило бы 4). Значит, G не имеет циклов.

Вторая часть условия 5) вытекает, согласно лемме 1.2, из того, что G - связный. Таким образом, получаем, что 4) \rightarrow 5). Пусть выполняется 5). Если при этом G - не связный граф и вершины u, v лежат в разных связных компонентах графа G , то добавление к G ребра (u, v) , очевидно, не порождает циклов, что противоречит 5). Отсюда следует, что G - связный граф, то есть 5) \rightarrow 1). Теорема 2.1 доказана.

Условия 4) и 5) показывают, что множество всех деревьев - это множество всех минимальных связных графов и, в то же время, множество всех максимальных графов без циклов.

Для представления данных в информационных системах, в справочниках, при реализации алгоритмов поиска и в других приложениях часто используются *корневые деревья*, то есть деревья с выделенной вершиной, именуемой корнем. Мы дадим следующее индуктивное определение корневого дерева.

1) Граф, имеющий одну вершину v , которая выделена, и не имеющий ребер, является корневым деревом с корнем v .

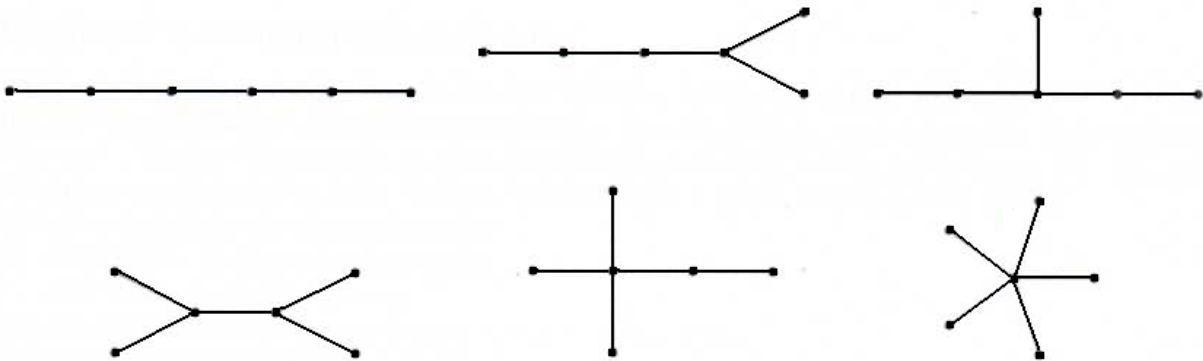
2) Пусть D_1, D_2, \dots, D_m , где $m \geq 1$ - корневые деревья с корнями v_1, v_2, \dots, v_m . Пусть $D_i = (V_i, E_i)$ и $V_i \cap V_j = \emptyset$ при $i \neq j$. Пусть $v \in V = V_1 \cup V_2 \cup V_3 \cup \dots \cup V_m$. Тогда граф $D = (V, E)$, где $V = V_1 \cup V_2 \cup V_3 \cup \dots \cup V_m \cup \{v\}$, $E = E_1 \cup E_2 \cup E_3 \cup \dots \cup E_m$ и выделена вершина v , является корневым деревом с корнем v .

3) Только такие графы называются корневыми деревьями, которые могут быть построены по 1) и 2).

Например, файловая структура в компьютере является корневым деревом. При этом корню соответствует сам компьютер, вершинам второго яруса соответствуют диски A, B, C, D и т.д., вершинам третьего яруса соответствуют директории, вершинам следующих ярусов соответствуют поддиректории и файлы.

Задачи и упражнения

1. На рисунке представлены все деревья (связные графы без циклов) на 6 вершинах. Для каждого из этих графов найдите группу симметрий, разбиения множества вершин на орбиты этой группы, стабилизаторы каждой вершины и фиксаторы некоторых пар вершин (на Ваш выбор).
2. Найдите группу симметрий
 - а) Полного графа на n вершинах;
 - б) цепи длины n ;
 - в) цикла длины n ;
 - г) каркаса куба.



3. Группа симметрий графа — вся группа перестановок его вершин. Что это за граф?
4. а) Приведите пример графа, не имеющего симметрий, кроме тождественной,
 б) Тот же вопрос при условии, что граф должен содержать цикл.
5. Пусть в графе на n вершинах любую из вершин можно перевести в любую с помощью симметрии. Сколько центров может быть у этого графа?
6. Группа симметрий графа — вся группа перестановок его вершин. Что это за граф?
7. а) Приведите пример графа, не имеющего симметрий, кроме тождественной,

б) Тот же вопрос при условии, что граф должен содержать цикл.

8. Пусть в графе на n вершинах любую из вершин можно перевести в любую с помощью симметрии. Сколько центров может быть у этого графа?

2.1. Остовные деревья

Существует несколько эквивалентных определений. Если $G = (X, A)$ – неориентированный граф с n вершинами, то остовным деревом (остовом) графа G называется всякий остовный подграф графа G , являющийся деревом. Например, для графа G , показанного на рис.2.1(а) графы на рис.2.1(б, в) являются остовами графа G . Остов графа G можно также рассматривать как минимальный связный остовный подграф графа G , где «минимальность» понимается в том смысле, что никакое собственное подмножество ребер этого остова не образует связный остовный подграф графа G .

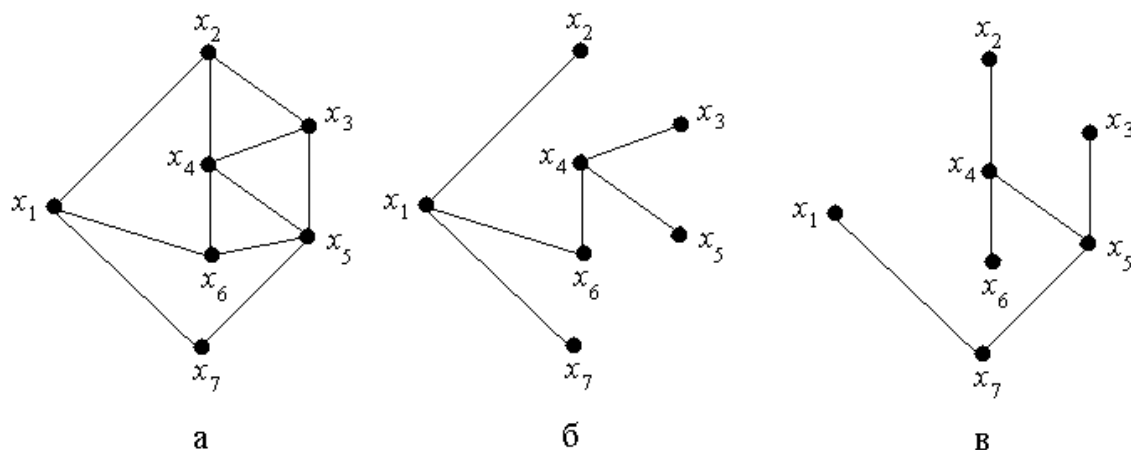


Рис.2.1. Граф G и его остовы.

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

На рис.2.2 показан граф, который является ориентированным деревом с корнем в вершине x_1 . Из приведенного определения следует, что ориентированное дерево с n вершинами имеет $n-1$ дуг и

связно. Также очевидно, что не всякий ориентированный граф содержит остовное ориентированное дерево.

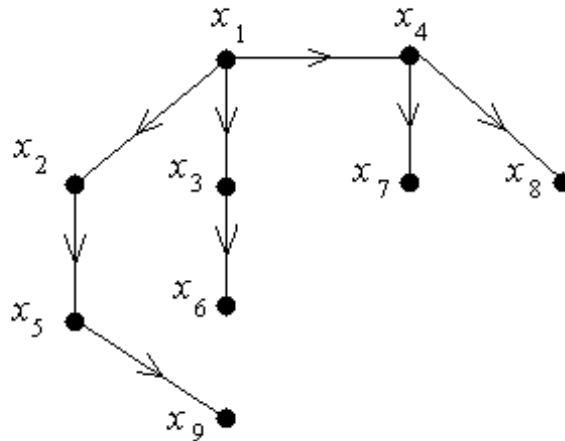


Рис.2.2. Ориентированное дерево.

Неориентированное дерево можно преобразовать в ориентированное: надо взять его произвольную вершину в качестве корня и ребрам приписать такую ориентацию, чтобы каждая вершина соединялась с корнем (только одной) простой цепью. Обратно, если $T = (X, A)$ - ориентированное дерево, то $\bar{T} = (X, \bar{A})$, где \bar{A} - множество дуг дерева T без учета их ориентации, является неориентированным деревом.

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

2.2. Построение всех остовных деревьев графа

В ряде случаев возникает необходимость в построении полного списка остовных деревьев графа G . В том случае, когда надо выбрать «наилучшее» дерево, а критерий, позволяющий осуществить такой отбор, является очень сложным, так что непосредственное решение задачи оптимизации (не использующее перечисление всех остовных деревьев) оказывается невыполнимым.

При анализе всякого алгоритма особое место занимает оценка его трудоемкости. Под трудоемкостью алгоритма будем понимать время выполнения соответствующей программы на ЭВМ.

Трудоемкостью алгоритма называется функция f , ставящая в соответствие каждому натуральному числу n время работы $f(n)$

алгоритма в худшем случае на входах длины n [2]. При анализе и обосновании алгоритмов будем использовать O - символику.

Будем говорить, что неотрицательная функция $f(n)$ не превосходит по порядку функцию $g(n)$, если существует такая константа c , что $f(n) \leq cg(n)$ для всех $n \geq 0$ и при этом будем писать $f(n) = O(g(n))$.

Иногда вместо предложения «трудоемкость алгоритма есть $O(g(n))$ » будем говорить «алгоритм решает задачу за время $O(g(n))$ ». Трудоемкость $O(1)$ означает, что время работы соответствующего алгоритма не зависит от n . Алгоритм, сложность которого равна $O(n^c)$, где c - константа, называется полиномиальным или алгоритмом полиномиальной сложности.

Под полиномиальной разрешимостью некоторой массовой задачи понимают существование полиномиального алгоритма, гарантирующего нахождение решения всякой ее индивидуальной задачи. Термин «класс задач» означает определенное семейство некоторых массовых задач.

Число различных остовов полного связного неориентированного помеченного графа с n вершинами равно n^{n-2} . Число различных остовов неориентированного графа без петель с n вершинами равно значению определителя $|A_0 \cdot A_0^t|$, где A_0 - матрица инциденций с одной удаленной строкой (т.е. с $n-1$ независимыми строками), A_0^t - транспонированная матрица к A_0 .

Рассмотрим два остовных дерева $T_1 = (X, A_1)$ и $T_2 = (X, A_2)$ графа $G = (X, A)$. «Расстояние» между двумя деревьями обозначается через $d(T_1, T_2)$ и определяется как число дуг из T_1 , которых нет в T_2 (или, что эквивалентно, как число дуг из T_2 , которых нет в T_1 , поскольку оба дерева T_1 и T_2 имеют $(n-1)$ дуг).

Если $d(T_1, T_2) = 1$, т.е. если $(A_1 \cup A_2) - (A_1 \cap A_2) = \{a_1, a_2\}$, где $a_1 \in A_1$ и $a_2 \in A_2$, то дерево T_2 можно получить из дерева T_1 удалив из T_1 дугу a_1 и добавив дугу a_2 . Такое преобразование дерева T_1 в дерево T_2 называется элементарным преобразованием дерева. На рис. 2.3 приведено дерево T_4 , которое получено из дерева T_0 с помощью 4 элементарных преобразований: убрать (x_1, x_8) и

добавить (x_5, x_8) ; удалить (x_5, x_6) и добавить (x_6, x_7) ; убрать (x_2, x_3) и добавить (x_2, x_4) ; удалить (x_3, x_5) и добавить (x_4, x_5) .

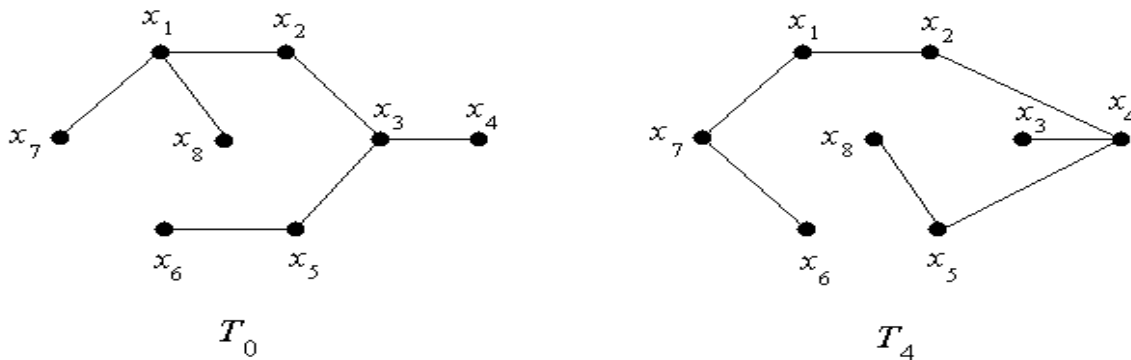


Рис.2.3. Остовы T_0 и T_4 .

Существует теорема, согласно которой дерево T_k может быть получено из дерева T_0 с помощью серий из k элементарных преобразований при $d(T_0, T_k) = k$.

2.3. Алгоритм нахождения всех деревьев графа

Первый шаг состоит в приписывании номеров ребрам графа G (ребра нумеруются от 1 до m , где m - число ребер в графе G). На каждом этапе (т. е. при каждом ветвлении в дереве решений) выбирается ребро, которое вместе с остальными, выбранными уже на предыдущих этапах, будет образовывать часть конструируемого дерева. Таким образом, прежде чем выбрать такое ребро, надо выяснить, действительно ли добавление его к частично сформированному дереву (которое на этом шаге является набором поддеревьев) не приводит к появлению цикла. Если цикл появляется, то данное ребро отбрасывается и проверяется следующее ребро, с большим номером. Если цикла нет, то ребро добавляется к другим, уже выбранным, и процесс продолжается до тех пор, пока не будет построено остовое дерево. Ребра перебираются в порядке возрастания их номеров; это приводит к исчерпывающему и без повторений решению задачи.

Для облегчения операций с поддеревьями в каждом поддереве выделяют произвольным образом корень (некоторую вершину поддерева) и затем рассматривается поддерево как древовидность. Для проверки на возможность образования цикла (при добавлении ребра) каждую вершину x_j помечают парой (r_j, p_j) . Первая метка r_j

указывает «корень» поддерева, содержащего вершину x_j . Первоначально $r_j = x_j$ для всех вершин x_j . На некотором шаге два поддерева T_1 , и T_2 сращиваются посредством добавления ребра $a_1=(x_\alpha, x_\beta)$ с вершиной x_α из T_1 и вершиной x_β из T_2 . Если на этом шаге r_1 - «корневая» метка вершин в T_1 , а r_2 - «корневая» метка вершин в T_2 и для примера $r_1 < r_2$, то все вершины в T_2 , должны «сменить» свои корневые метки на r_1 , и два поддерева T_1 и T_2 «сольются» в единственное новое дерево T_1 .

Вторая метка p_i , приписанная вершине x_j , указывает вершину, предшествующую вершине x_j , т.е. если (x_k, x_j) - дуга рассматриваемого поддерева, то $p_j = x_k$. Для корневой вершины дерева такая метка полагается равной нулю.

1. Замена корня дерева

Если корнем дерева T является вершина r и нужно в качестве корня выбрать новую вершину x_s , то такую «замену» r на x_s можно осуществить простым обращением ориентации дуг, принадлежащих цепи, идущей от r к x_s , не меняя при этом ориентацию других дуг. Соответствующие изменения меток «предшествования» будут таковы.

1. Пусть $x_j = x_s$ и $z = p_j$.
2. Положить $x_i = z$, а $z = p_i$.
3. Шаг обновления: $p_i = x_j$.
4. Если $x_i = r$, то перейти к шагу 5. в противном случае положить $x_j = x_i$ и перейти к шагу 2.
5. Положить $p_s = 0$, стоп.

Все вершины, имеющие корневую метку r , заменить ее на метку x_s .

3. Сращивание двух поддереьев

Если осуществлено сращивание двух поддереьев T_1 и T_2 (путем добавления ребра (x_α, x_β)), то метки необходимо изменить:

- (1) У вершин с корневой меткой r_2 заменить эту метку на r_1 .
- (2) Заменить в дереве T_2 корень r_2 на x_β (пункт А), после чего изменить метку «предшествования» у вершины x_β с $p_\beta = 0$ на $p_\beta = x_\alpha$.

3. Расщепление дерева на две части

Поскольку метод порождения деревьев, рассматриваемый ниже, является поиском, использующим дерево решений, то возникает необходимость удаления некоторых ребер (на шагах возвраще-

ния), чтобы испытать затем другие ребра. В такой ситуации удаление ребра приводит к расщеплению некоторого дерева на две части, например на T_1 и T_2 , и метки одного из этих поддеревьев должны быть изменены.

Пусть удаляется ребро (x_α, x_β) , где $x_\alpha \in T_1$ и $x_\beta \in T_2$. Тогда при $p_\beta = x_\alpha$ (т. е. если ребро (x_α, x_β) в первоначальном дереве ориентировано от x_α к x_β) метки в дереве T_1 , можно оставить прежними, а метки в дереве T_2 должны быть изменены.

Если же $p_\alpha = x_\beta$ (т. е. ребро (x_α, x_β) ориентировано от x_β к x_α), то можно не менять метки в дереве T_2 , но нужно изменить метки в T_1 . Предполагая, что метки меняются в дереве T_2 , покажем, как надо «восстанавливать» корень в этом дереве.

1. Положить $S = \{x_\beta\}$ и $p_\beta = 0$ (x_β будет корнем дерева T_2).

2. Найти все вершины x_j с $p_j \in S$ и изменить их корневые метки на $r_j = x_\beta$. Если таких вершин нет, то остановиться.

3. Шаг обновления: $S = S \cup \{x_j / p_j \in S\}$; вернуться к шагу 2.

Следует отметить, что ни у одной вершины, кроме нового корня x_β , метки предшествования менять не нужно. Заметим также, что число описанных выше шагов 2 и 3, которое необходимо для восстановления корня, равно длине самой длинной цепи в T_2 исходящей из вершины x_β .

Описание алгоритма.

Возьмем произвольную вершину x^* графа G . Пусть ее степень равна d^* . Перенумеруем ребра, инцидентные этой вершине: a_1, a_2, \dots, a_{d^*} . Затем перенумеруем остальные ребра графа G : $a_{d^*+1}, a_{d^*+2}, \dots, a_m$. При порождении деревьев ребра будут перебираться в соответствии с введенной нумерацией.

Шаг 1. Приписать вершинам метки: (r_i, p_i) , где $r_i = x_i$ и $p_i = 0$, $\forall x_i \in X$. Положить $k=1$.

Шаг 2. Выбрать для исследования некоторое ребро. Например, $a_k = (x_i, x_j)$. Если $k \leq m$, где m — число ребер графа, то перейти к 2 (1). При $k = m+1$, т. е. если «неисследованных» ребер нет, перейти к шагу 5.

(1) Если $r_i = r_j$, то это означает, что вершины x_i и x_j принадлежат одному и тому же поддереву и добавление ребра a_k приведет к появлению цикла. Отбросить ребро a_k , т. е. положить $k = k-1$ и вернуться к шагу 2.

(2) Если $r_i \neq r_j$ то ребро a_k можно добавить к ребрам построенных поддеревьев. Перейти к шагу 3.

Шаг 3. Срастить два поддерева, у которых вершины имеют корневые метки r_i и r_j , применив для этого метод, описанный выше в пункте 2.

Шаг 4. Выбрав $n-1$ ребер, мы получаем некоторое дерево. Заполнить это дерево и перейти к шагу 5. Если выбрано меньше, чем $n-1$ ребер, то положить $k=k+1$ и вернуться к шагу 2.

Шаг 5. (Возвращение.) Удалить ребро, добавленное последним. Предположим, что таким ребром является a_l . Если a_l - единственное оставшееся для добавления ребра $l=d^*$, то остановиться. Все остовные деревья, таким образом, построены, так как при любом дальнейшем ветвлении дерева решений вершина x^* останется изолированной. В противном случае надо обновить метки, действуя так, как указано в пункте 2, положить $k=l-1$ и возвратиться к шагу 2.

Пример 2.1. Построить все остовные деревья графа, изображенного на рис.2.4.

Выберем в качестве x^* вершину x_1 .

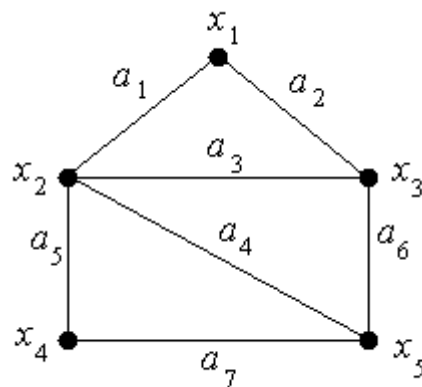


Рис.2.4. Граф G.

На рис.2.5 изображено соответствующее дерево решений, которое порождено в процессе работы алгоритма. Если выделить ребра, указанные в кружочках какой-либо цепи, выходящей из верхнего узла этого дерева и оканчивающейся в самом нижнем узле, то из них можно построить некоторый остов данного графа. Эти остовы перенумерованы числами от 1 до 21 и приведены на рис.2.6.

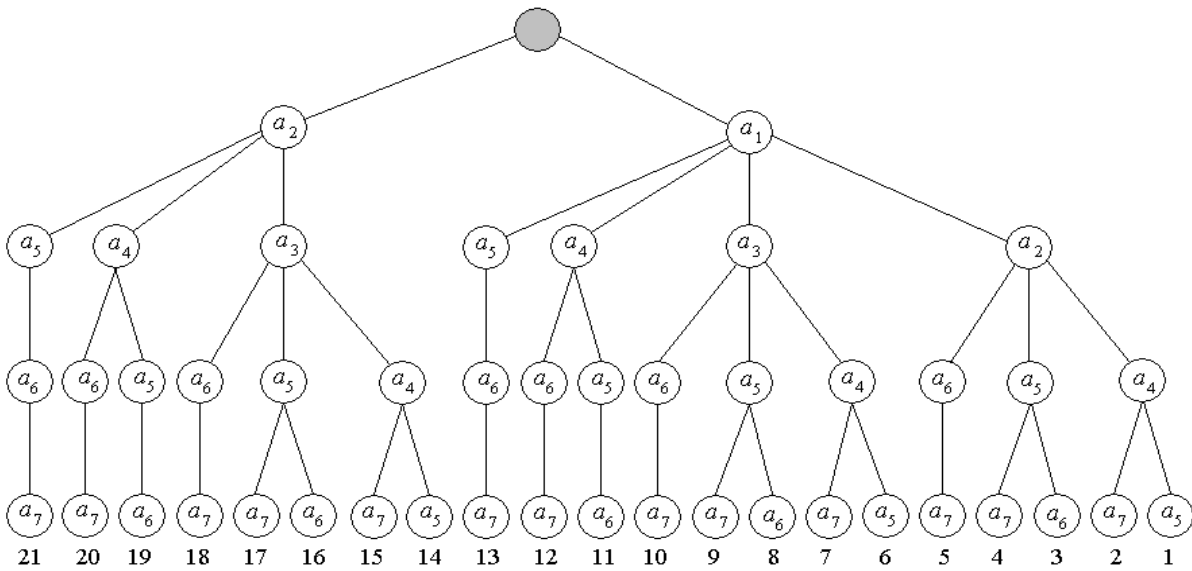


Рис.2.5. Полное дерево поиска.

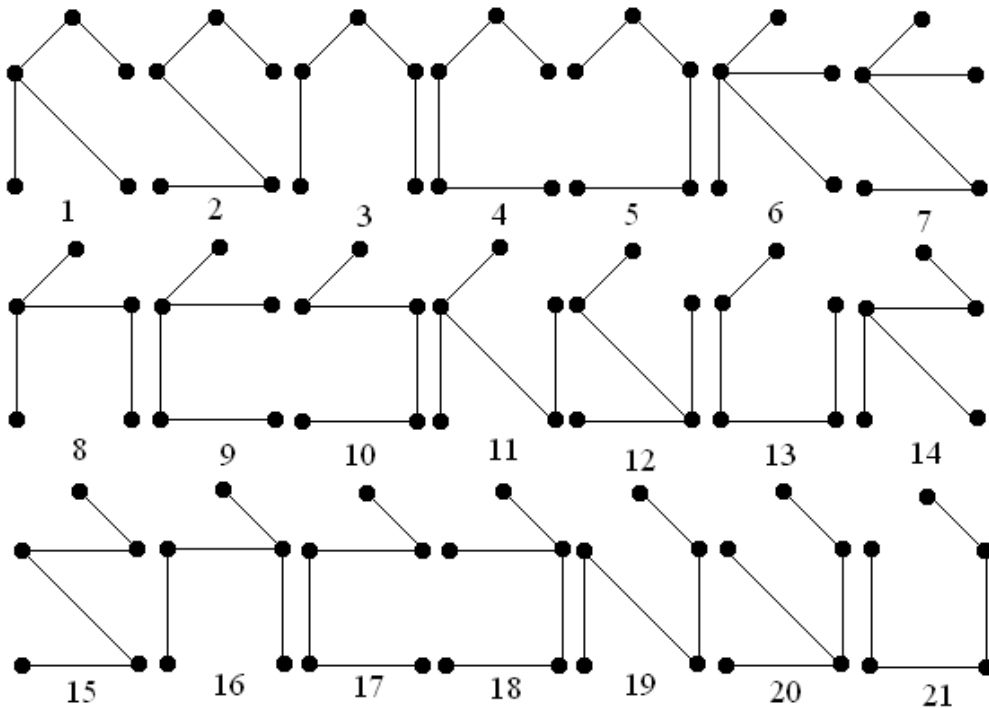


Рис.2.6. Все остовы графа G.

2.4. Алгоритмы построения остовного дерева минимального веса графа

Рассмотрим взвешенный связный неориентированный граф $G=(X, A)$; вес ребра (x_i, x_j) обозначим c_{ij} . Из большого числа остовов графа нужно найти один, у которого сумма весов ребер наименьшая. Такая задача возникает, например, в том случае, когда вершины являются клеммами электрической сети, которые

должны быть соединены друг с другом с помощью проводов наименьшей общей длины (для уменьшения уровня наводок). Другой пример: вершины представляют города, которые нужно связать сетью трубопроводов; тогда наименьшая общая длина труб, которая должна быть использована для строительства (при условии, что вне городов «разветвления» трубопроводов не допускаются), определяется ОДМВ (остовным деревом минимального веса) или кратчайшим остовом соответствующего графа.

Следует отметить, что кратчайший остов графа не имеет никакого отношения к дереву, дающему все кратчайшие пути, выходящие из некоторой выбранной вершины.

Задача построения кратчайшего остова графа является одной из немногих задач теории графов, которые можно считать полностью решенными. Итак, пусть T_i и T_j — два произвольных поддерева, полученных путем добавления ребер при построении кратчайшего остова графа. Определим Δ_{ij} , как кратчайшее из расстояний между вершинами из T_i и вершинами из T_j следующим образом:

$$\Delta_{ij} = \min_{x_i \in T_i} [\min_{x_j \in T_j} \{c(x_i, x_j)\}], i \neq j. \quad (2.1)$$

Поставим следующую задачу: для поддерева T_s найти такое поддерево T_{j^*} , чтобы $\Delta_{sj^*} = \min_{T_j} [\Delta_{sj}]$. Пусть (x_s, x_{j^*}) - ребро, вес которого соответствует величине Δ_{sj^*} в выражении (2.1). Тогда ребро (x_s, x_{j^*}) принадлежит кратчайшему остову и может быть добавлено к другим ребрам частично сформированного кратчайшего остова.

Многочисленное применение нижеописанной операции приводит к построению кратчайшего остова графа. Многие методы, позволяющие строить кратчайший остов графов, основываются на частных случаях описанной выше операции. Первый из таких методов был предложен Краскалом.

Алгоритм Краскала

В этом алгоритме для добавления к частично сформированному дереву выбирается абсолютно кратчайшее допустимое ребро, а не просто кратчайшее ребро между одним поддеревом и другим каким-либо поддеревом (как это предполагалось в приведенной выше операции). Так как выбранное ребро является кратчайшим

между некоторым поддеревом и каким-то другим поддеревом, то правило выбора в этом алгоритме представляет собой частный случай операции.

Алгоритм имеет следующий вид:

Шаг 1. Начать с вполне несвязного графа T , содержащего n вершин.

Шаг 2. Упорядочить ребра графа G в порядке неубывания их весов.

Шаг 3. Начав с первого ребра в этом списке, добавлять ребра в графе T , соблюдая условие: такое добавление не должно приводить к появлению цикла в T .

Шаг 4. Повторять шаг 3 до тех пор, пока число ребер в T не станет равным $n-1$. Получившееся дерево является кратчайшим остовом графа G .

При выполнении этого алгоритма может возникнуть такая ситуация, когда очередное кратчайшее ребро, выбранное из списка, построенного на шаге 2, будет соединять две вершины одного и того же поддерева. Добавлять это ребро к T нельзя, поскольку такое добавление приводит к появлению цикла в T . Поэтому на шаге 3, прежде чем добавить ребро к графу T , надо проверить, является ли оно допустимым в указанном выше смысле. Такую проверку можно выполнить более эффективно (путем осуществления одного сравнения с использованием процедуры расстановки меток из раздела 2.2), абсолютно так же, как это делается на шаге 2 алгоритма из раздела 2.2. Известно, что трудоемкость алгоритма Краскала составляет $O(n^3)$ операций.

Алгоритм Прима

Этот алгоритм порождает кратчайший остов посредством увеличения только одного поддерева T_s , содержащего больше одной вершины. «Одиночные» вершины рассматриваются как отдельные поддерева. Поддерево T_s постепенно увеличивается за счет присоединения ребер (x_i, x_j) , где $x_i \in T_s$ и $x_j \notin T_s$; причем добавляемое ребро должно иметь наименьший вес c_{ij} . Процесс продолжается до тех пор, пока число ребер в T_s не станет равным $n-1$. Тогда поддерево T_s будет требуемым кратчайшим остовом. Впервые такая частная форма операции была предложена Прима.

Алгоритм начинает работу с присвоения каждой вершине $x_j \notin T_s$ метки $[\alpha_j, \beta_j]$, где α_j на каждом шаге есть ближайшая к x_j верши-

на из поддеревя T_s , а β_j - вес ребра (α_j, x_j) . На каждом шаге выполнения алгоритма вершина, например x_{j^*} , с наименьшей меткой β_j присоединяется к T_s посредством добавления ребра (α_{j^*}, x_{j^*}) . Поскольку к T_s добавлена новая вершина x_{j^*} , то, может быть, придется изменить метки $[\alpha_j, \beta_j]$ у некоторых вершин $x_j \notin T_s$ (если, например, $c(x_j, x_{j^*})$ меньше существующей метки β_j) и после этого продолжить процесс.

Алгоритм имеет следующий вид:

Шаг 1. Пусть $T_s = \{x_s\}$, где x_s — произвольно выбранная вершина, и $A_s = \emptyset$ (A_s является множеством ребер, входящих в кратчайший остов).

Шаг 2. Для каждой вершины $x_j \notin T_s$ найти вершину $\alpha_j \in T_s$, такую, что $c(\alpha_j, x_j) = \min_{x_i \in T_s} [c(x_i, x_j)] = \beta_j$,

и приписать вершине x_j метку $[\alpha_j, \beta_j]$. Если такой вершины α_j нет, т.е. при $\Gamma(x_j) \cap T_s = \emptyset$, приписать вершине x_j метку $[0, \infty]$.

Шаг 3. Выбрать такую вершину x_{j^*} , что $\beta_{j^*} = \min_{x_j \notin T_s} [\beta_j]$.

Обновить данные: $T_s = T_s \cup \{x_{j^*}\}$, $A_s = A_s \cup \{(\alpha_{j^*}, x_{j^*})\}$.

Если $|T_s| = n$, то остановиться.

Ребра в A_s образуют кратчайший остов.

Если $|T_s| \neq n$, то перейти к шагу 4.

Шаг 4. Для всех $x_j \notin T_s$, таких, что $x_j \in \Gamma(x_{j^*})$, обновить метки следующим образом. Если $\beta_j > c(x_{j^*}, x_j)$, то положить $\beta_j = c(x_{j^*}, x_j)$, $\alpha_j = x_{j^*}$; перейти к шагу 3. Если $\beta_j \leq c(x_{j^*}, x_j)$, то перейти к шагу 3.

Известно, что трудоемкость алгоритма Прима составляет $O(n^2)$ операций.

Пример 2.2. На графе G , изображенном на рис.2.7, каждая вершина представляет некоторое лицо, а ребра (x_i, x_j) отражают тот факт, что лицо x_i может общаться с лицом x_j и наоборот.

Требуется определить такой способ передачи конфиденциального сообщения между 12 лицами, при котором вероятность утечки информации будет наименьшей. Каждой передаче сообщения от x_i к x_j приписывается некоторая вероятность p_{ij} того, что послание может быть перехвачено посторонним лицом. Эти вероятности в процентах даны на рис. 2.7. Очевидно, что пути передачи сообщения должны образовывать остовное дерево графа G , и требуется найти такое остовное дерево, которое минимизирует величину

$1 - \prod_{(x_i, x_j) \in T} (1 - p_{ij})$, где произведение берется по всем ребрам этого

дерева. Поскольку эта функция возрастающая и симметричная относительно p_{ij} , то требуемое остовное дерево будет совпадать с кратчайшим остовом графа G , при этом p_{ij} принимаются за «стоимости» ребер c_{ij} .

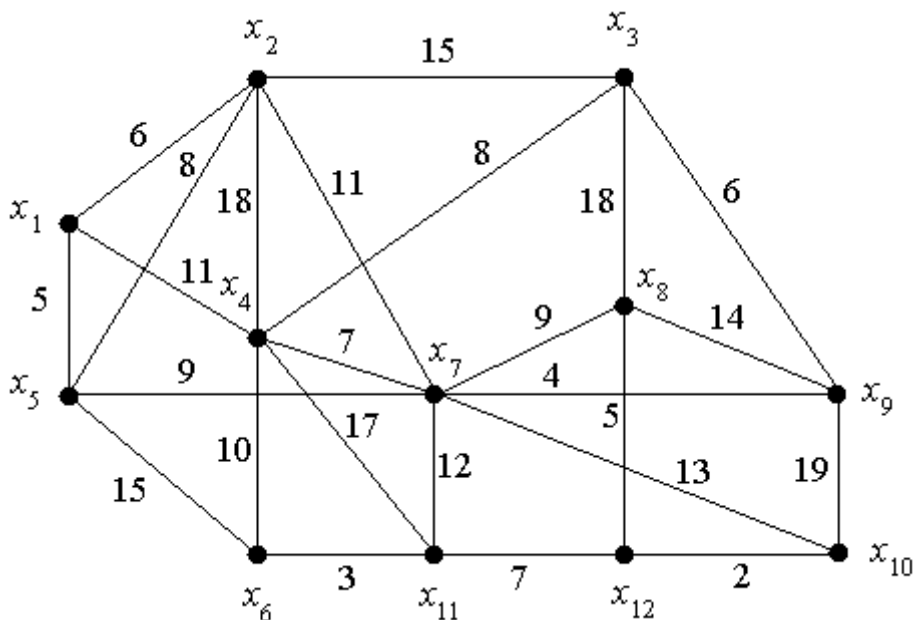


Рис.2.7. Граф G из примера 2.3.

Для решения этой задачи применим алгоритм Прима.

Шаг 1. Возьмем $x_5 = x_1$, $T_1 = \{x_1\}$, $A_1 = \emptyset$.

Шаг 2. Примем за метки для x_2 , x_4 и x_5 : $[x_2, 6]$, $[x_4, 11]$ и $[x_5, 5]$ соответственно. Остальные метки равны $[0, \infty]$.

Шаг 3. Наименьшая метка β_j будет у вершины x_5 , и поскольку $\alpha_5 = x_1$ то построим ребро (x_1, x_5) : $T_2 = \{x_1, x_5\}$, $A_2 = \{(x_1, x_5)\}$.

Шаг 4. Обновим метки вершин x_2 , x_6 , x_7 следующим образом:

для x_2 : $\beta_2 = 6 \leq c(x_5, x_2)$ и обновления не требует;

для x_6 : $\beta_6 = \infty > c(x_5, x_6) = 15$, следовательно, метка для x_6 примет вид $[x_5, 15]$;

для x_7 : $\beta_7 = \infty > c(x_5, x_7) = 9$, следовательно, метка для x_7 примет значение $[x_5, 9]$.

Поскольку $x_4 \notin \Gamma(x_5)$, ее метка остается неизменной.

Шаг 3. Метки теперь таковы: для x_2 : $[x_1, 6]$, для x_4 : $[x_1, 11]$, для x_6 : $[x_5, 15]$, для x_7 : $[x_5, 9]$. Наименьшая метка β_j будет у x_2 , и поскольку $\alpha_2 = x_1$, то построим ребро (x_1, x_2) : $T_3 = \{x_1, x_5, x_2\}$, $A_3 = \{(x_1, x_5), (x_1, x_2)\}$.

Шаг 4. Обновим метки вершин x_3, x_4, x_7 следующим образом:

для x_3 : $[x_2, 15]$;

для x_4 : $[x_1, 11]$ (метка не обновляется);

для x_7 : $[x_5, 9]$ (метка не обновляется).

Поскольку $x_6 \notin \Gamma(x_2)$, ее метка остается неизменной.

Шаг 3. Наименьшая метка β_j будет у вершины x_7 , и поскольку $\alpha_7 = x_5$ то построим ребро (x_5, x_7) : $T_4 = \{x_1, x_5, x_2, x_7\}$, $A_4 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7)\}$.

Шаг 4. Обновим метки вершин $x_8, x_9, x_{10}, x_{11}, x_4$ следующим образом:

для x_8 : $[x_7, 9]$;

для x_9 : $[x_7, 4]$;

для x_{10} : $[x_7, 13]$;

для x_{11} : $[x_7, 12]$;

для x_4 : $[x_7, 7]$.

Поскольку $x_3 \notin \Gamma(x_7)$ и $x_6 \notin \Gamma(x_7)$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_9 , и поскольку $\alpha_9 = x_7$ то построим ребро (x_7, x_9) : $T_5 = \{x_1, x_5, x_2, x_7, x_9\}$, $A_5 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9)\}$.

Шаг 4. Обновим метки вершин x_3, x_8, x_{10} следующим образом:

для x_3 : $[x_9, 6]$;

для x_8 : $[x_7, 9]$ (метка не обновляется);

для x_{10} : $[x_7, 13]$ (метка не обновляется).

Поскольку $x_4 \notin \Gamma(x_9)$, $x_6 \notin \Gamma(x_9)$ и $x_{11} \notin \Gamma(x_9)$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_3 , и поскольку $\alpha_3 = x_9$ то построим ребро (x_9, x_3) : $T_6 = \{x_1, x_5, x_2, x_7, x_9, x_3\}$, $A_6 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3)\}$.

Шаг 4. Обновим метки вершин x_4, x_8 следующим образом:

для x_4 : $[x_7, 7]$ (метка не обновляется);

для x_8 : $[x_7, 9]$ (метка не обновляется).

Поскольку $x_6 \notin \Gamma(x_3)$, $x_{10} \notin \Gamma(x_3)$ и $x_{11} \notin \Gamma(x_3)$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_4 , и поскольку $\alpha_4 = x_7$ то построим ребро (x_7, x_4) : $T_7 = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4\}$, $A_7 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4)\}$.

Шаг 4. Обновим метки вершин x_6, x_{11} следующим образом:

для x_6 : $[x_4, 10]$;

для x_{11} : $[x_7, 12]$ (метка не обновляется).

Поскольку $x_8 \notin \Gamma(x_4)$ и $x_{10} \notin \Gamma(x_4)$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_8 , и поскольку $\alpha_8 = x_7$ то построим ребро (x_7, x_8) : $T_8 = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4, x_8\}$, $A_8 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4), (x_7, x_8)\}$.

Шаг 4. Обновим метки вершины x_{12} следующим образом:

для x_{12} : $[x_8, 5]$.

Поскольку $x_6 \notin \Gamma(x_8)$, $x_{10} \notin \Gamma(x_8)$ и $x_{11} \notin \Gamma(x_8)$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_{12} , и поскольку $\alpha_{12} = x_8$ то построим ребро (x_8, x_{12}) : $T_9 = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4, x_8, x_{12}\}$, $A_9 = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4), (x_7, x_8), (x_8, x_{12})\}$.

Шаг 4. Обновим метки вершин x_{10} , x_{11} следующим образом:

для x_{10} : $[x_{12}, 2]$;

для x_{11} : $[x_{12}, 7]$.

Поскольку $x_6 \notin \Gamma(x_{12})$, ее метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_{10} , и поскольку $\alpha_{10} = x_{12}$ то построим ребро (x_{12}, x_{10}) : $T_{10} = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4, x_8, x_{12}, x_{10}\}$, $A_{10} = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4), (x_7, x_8), (x_8, x_{12}), (x_{12}, x_{10})\}$.

Шаг 4. Обновлять метки смежных вершин не надо.

Поскольку $x_6 \notin \Gamma(x_{10})$ и $x_{11} \notin \Gamma(x_{10})$, их метки остаются неизменными.

Шаг 3. Наименьшая метка β_j будет у вершины x_{11} , и поскольку $\alpha_{11} = x_{12}$ то построим ребро (x_{12}, x_{11}) : $T_{11} = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4, x_8, x_{12}, x_{10}, x_{11}\}$, $A_{11} = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4), (x_7, x_8), (x_8, x_{12}), (x_{12}, x_{10}), (x_{12}, x_{11})\}$.

Шаг 4. Обновим метки вершины x_6 следующим образом:

для x_6 : $[x_{11}, 3]$.

Шаг 3. Наименьшая метка β_j будет у вершины x_6 , и поскольку $\alpha_6 = x_{11}$ то построим ребро (x_{11}, x_6) : $T_{12} = \{x_1, x_5, x_2, x_7, x_9, x_3, x_4, x_8, x_{12}, x_{10}, x_{11}, x_6\}$, $A_{12} = \{(x_1, x_5), (x_1, x_2), (x_5, x_7), (x_7, x_9), (x_9, x_3), (x_7, x_4), (x_7, x_8), (x_8, x_{12}), (x_{12}, x_{10}), (x_{12}, x_{11}), (x_{11}, x_6)\}$.

Так как $|T_{12}| = 12$, то следует остановиться. Ребра в A_{12} образуют кратчайший остов. Кратчайший остов графа G приведен на рис.2.8, на котором порядок нумерации ребер соответствует последовательности их включения в дерево.

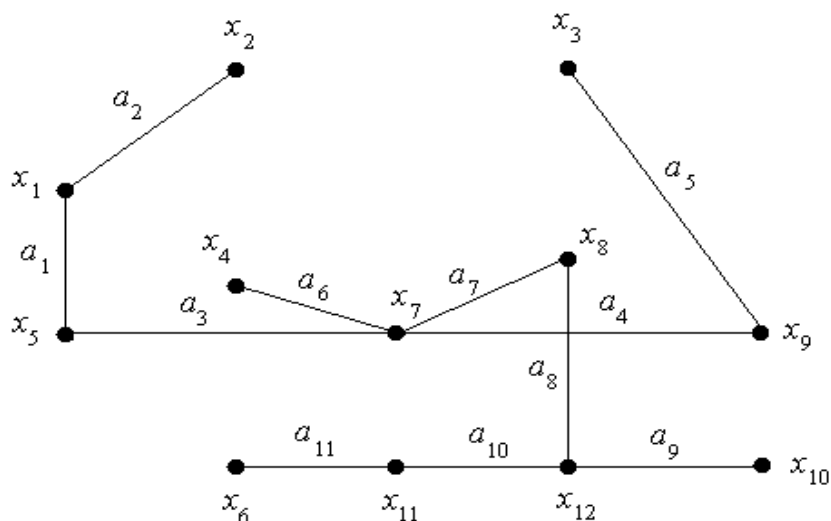


Рис. 2.8. Кратчайший остов графа G на рис.2.7.

Произведение $\prod(1 - \rho_{ij})$ для ребер этого дерева равно 0.5214, и, следовательно, величина минимума вероятности перехвата сообщения посторонним лицом равна 0.4786.

Задачи и упражнения

1. Найдите все попарно неизоморфные деревья
 - а) не более, чем на 7 вершинах;
 - б) на 9 вершинах, в которых есть путь длины 7;
 - в) на 9 вершинах, в которых есть две вершины степени 4;
 - г) на 9 вершинах, в которых есть вершина степени 6.
2. На доске нарисовано 7 графов, каждый из которых — дерево с 6 вершинами. Докажите, что среди них есть два изоморфных.
3. Докажите, что
 - а) граф, в котором любые две вершины соединены ровно одним простым путем — дерево;
 - б) и, наоборот, в дереве любые две вершины соединены ровно одним простым путем.
 - в) при удалении любого ребра дерева получится несвязный граф.
4. а) Докажите, что в дереве есть вершина, из которой выходит ровно одно ребро (такая вершина называется висячей).
 - б) Докажите, что любое дерево содержит не менее двух висячих вершин.
 - в) Существует единственное дерево, для которого оба утверждения не верны. Что это за дерево? Опишите все деревья с

- г) двумя;
 - д) тремя висячими вершинами.
5. а) Удалим из дерева висячую вершину вместе с исходящим из нее ребром. Докажите, что граф останется деревом, но вершин и ребер в нем станет на одно меньше.
 - б) Докажите, что в любом дереве число ребер на 1 меньше числа вершин.
 6. И наоборот, докажите, что связный граф, у которого число ребер на единицу меньше числа вершин - дерево.
 7. В графе 10 вершин и 9 ребер. Верно ли, что этот граф - дерево?
 8. Существует ли дерево на 9 вершинах, в котором 2 вершины имеют степень 5?
 9. В дереве на 8 вершинах три вершины имеют степень 1. Сколько вершин имеют степень 3?
 10. Пусть T - дерево, в котором каждая вершина имеет степень 1 или k . Оценить количество вершин в дереве T .
 11. Докажите, что в дереве на $2n$ вершинах можно выбрать n вершин между которыми нет ни одного ребра.
 12. Пусть G граф с n вершинами ($n > 3$). такой, что каждый граф, получающийся из него удалением одной вершины — дерево. Что это за граф?
 13. Волейбольная сетка имеет вид прямоугольника размером 50 x 600 клеток. Какое наибольшее число веревочек можно перерезать так, чтобы сетка не распалась на куски?
 14. Составить программу нахождения ОДМВ графа (алгоритм Прима)

3. МЕТРИЧЕСКИЕ ХАРАКТЕРИСТИКИ ГРАФА

3.1. Эксцентриситет, радиус и диаметр

Чередующая последовательность

$$v_0, e_1, v_1, e_2, v_2, e_3, \dots, e_k, v_{k+1} \quad (3.1)$$

вершин и ребер графа $G=(V,E)$, начинающаяся в вершине и заканчивающаяся в вершине такая, что $v_0 \neq v_{k+1}$, называется маршрутом (или (v_0, v_{k+1}) – маршрутом). Маршрут (1) можно задать последовательностью

$$v_0, v_1, v_2, \dots, v_{k+1} \quad (3.2)$$

его вершин, а также последовательностью

$$e_1, e_2, e_3, \dots, e_k$$

его ребер.

Маршрут называется цепью, если все его ребра различны и простой цепью, если вершины кроме, возможно, крайних различны. Маршрут называется циклическим, если $v_0 = v_{k+1}$. Циклическая цепь называется циклом, а циклическая простая цепь – простым циклом. Число ребер в маршруте (3.1) называется его длиной.

Пусть $G=(V,E)$ – связный граф, а v_i, v_j – две его не совпадающие вершины. Длина кратчайшего (v_i, v_j) – маршрута называется расстоянием между вершинами v_i, v_j и обозначается через $d(v_i, v_j)$. Положим $d(v_i, v_j) = 0$, тогда и только тогда, когда $v_i = v_j$. Очевидно, что введенное таким образом расстояние удовлетворяет аксиомам метрики:

1. $d(v_i, v_j) \geq 0$

2. $d(v_i, v_j) = 0$, тогда и только тогда, когда $v_i = v_j$

3. $d(v_i, v_j) = d(v_j, v_i)$

4. $d(v_i, v_j) \leq d(v_j, v_k) + d(v_k, v_j)$

Пусть задан связный неориентированный граф $G=(V,E)$, $V=\{v_1, v_2, \dots, v_n\}$, $E=\{e\}$, $e=(v_i, v_j)$, $i, j=1, 2, 3, \dots, n$, $i \neq j$. Длина кратчайшего (v_i, v_j) –маршрута (простой цепи) называется расстоянием между вершинами v_i, v_j и обозначается через $d(v_i, v_j)$,

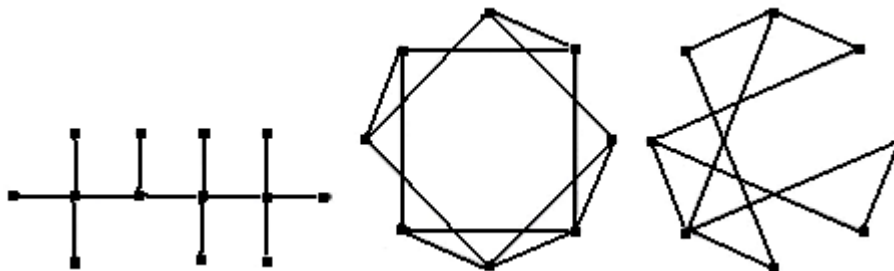
где $d(v_i, v_j) \neq 0$. Для фиксированной вершины v_{i_0} величина $e(v_{i_0}) = \max_{v_j \in V} d(v_{i_0}, v_j)$ называется эксцентриситетом этой вершины.

Максимальный среди всех эксцентриситетов вершин называется диаметром графа и обозначается через $d(G)$. Тем самым $d(G) = \max_{v_i \in V} e(v_i)$. Минимальный среди всех эксцентриситетов вершин графа G называется его радиусом и обозначается через $r(G) = \min_{v_i \in V} e(v_i)$. Задача состоит в нахождении радиуса и диаметра связного графа $G=(V, E)$.

Вершина v_i называется периферийной, если $e(v_i) = D(G)$. Вершина v_i называется центральной, если $e(v_i) = r(G)$.

Задачи и упражнения

1. Найдите диаметр, радиус и центры: а) пути б) цикла длины n .
2. Для графов на рисунке построить матрицу расстояний, найти максимальные удаления, диаметр, радиус и центры.



3. Докажите, что расстояние между вершинами графа удовлетворяет аксиомам метрики

- а) $\rho(x, x) = 0$
- б) $\rho(x, y) = \rho(y, x)$;
- в) $\rho(x, z) \leq \rho(x, y) + \rho(y, z)$ (неравенство треугольника).

4. Доказать, что для графа выполняется соотношение $r \leq d \leq 2r$.

Возведем матрицу смежности графа в квадрат.

- а) Докажите, что при $i \neq j$ в i -й строке и j -м столбце результата будет стоять число путей длины 2 из i -й вершины в j -ю;
- б) Чему будут равны числа на диагонали этой матрицы?
- в) Как аналогичным способом подсчитать число циклов длины 3 в графе?
- г) А путей длины 4?

5.Продолжая рассуждение, казалось бы, можно получить простую формулу для числа циклов длины n , т.е. для проверки гамильтоновости графа. Объясните, почему это неверно.

3.2. Двудольные графы. Теорема Кенига

Граф $G=(V,E)$ называется двудольным, если существует такое разбиение множества V его вершин на два подмножества V_1, V_2 (доли), что концы каждого ребра принадлежат различным долям и обозначаются $G=(V_1, V_2, E)$.

Теорема 3.1. (Кениг) Для двудольности графа G необходимо и достаточно, чтобы он не содержал циклов нечетной длины.

Доказательство. $1 \Rightarrow$. Необходимость. Предположим, что граф G – двудольный и C – один из его циклов длины k . Пройдем все ребра этого цикла в той последовательности, в которой они на нем расположены начиная с некоторой вершины v . Так как концы каждого ребра лежат в разных долях, то k -четно.

$2. \Rightarrow$ Достаточность. Не ограничивая общности можно рассматривать только связные графы, ибо дизъюнктивное объединение двудольных графов также двудольно. Пусть связный граф G порядка $n > 1$ не имеет циклов нечетной длины, $v \in V$.

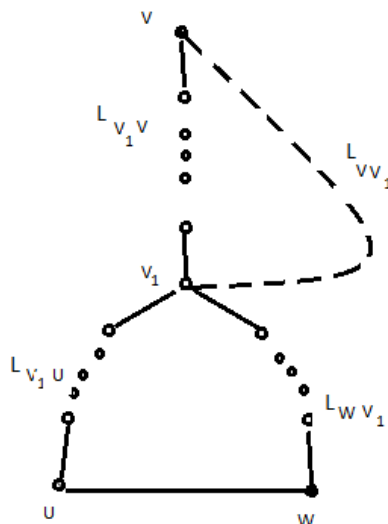


Рис.3.1. К доказательству теоремы Кёнига.

Построим разбиение V_1, V_2 следующим образом: произвольную вершину $u \in V$ отнесем к классу (множеству) V_1 , если расстояние $d(u,v)$ - четное число и к множеству V_2 , если $d(u,v)$ - нечетное число. Остается доказать, что порожденные подграфы $\langle V_1 \rangle, \langle V_2 \rangle$ являются пустыми, т.е. $\langle V_1 \rangle = \emptyset, \langle V_2 \rangle = \emptyset$.

Предположим, что один из графов $\langle V_1 \rangle$ не пуст. Тогда существуют две смежные вершины u, w входящие в один из графов $\langle V_1 \rangle$. Отсюда следует, что ни одна из них не совпадает с v , поскольку $v \in V_1$, а окружение v множеству V_2 . Пусть L_{uv} кратчайшая (u, v) - цепь, L_{wv} - кратчайшая (w, v) - цепь, v_1 последняя считая от v , из общих вершин этих цепей, лежащая на L_{uv} . Обозначим через L_{vv_1} и L_{v_1u} подцепи цепи L_{uv} , а через L_{vv_1} и L_{wv_1} подцепи цепи L_{wv} . Очевидно, что цепи $L_{vv_1} = L_{v_1v}$ и, следовательно, длины цепей L_{v_1u} и L_{wv_1} одного характера четности. Но тогда $L_{v_1u} \cup L_{wv_1} \cup \{(u, v)\}$ - цикл нечетной длины. Получили противоречие. Таким образом, достаточность, а вместе с ней и теорема 3.1 доказана.

3.3. Алгоритм поиска в ширину

Доказательство теоремы Кенига подсказывает простой способ распознавания двудольности графа. Этот способ основан на алгоритме поиска в ширину (ПШ). Алгоритм поиска в ширину приписывает вершинам графа номера $0, 1, 2, \dots, n$.

Алгоритм поиска в ширину начинает свою работу с произвольной вершины - v_0 и ей приписывает 0. Каждой вершине из окружения вершины 0 алгоритм поиска в ширину приписывает метку 1 и каждой нумерованной вершине из этого окружения, приписывает номер 2, и т.д. Если G - связный, то алгоритм поиска в ширину занумерует все вершины графа G . Далее разобьем V на два подмножества V_1 и V_2 , отнеся к V_1 вершины с четными номерами, а к V_2 - с нечетными номерами. Если при этом вершинно - порожденные графы пусты, т.е. $\langle V_1 \rangle = \emptyset$, $\langle V_2 \rangle = \emptyset$, то граф G является двудольным. Иначе граф G не является двудольным.

Рассмотрим порожденные графы $G(V_1)$ и $G(V_2)$. Если $G(V_1)$ и $G(V_2)$ - пусты, то $G = (V_1, V_2, E)$ - двудольный граф.

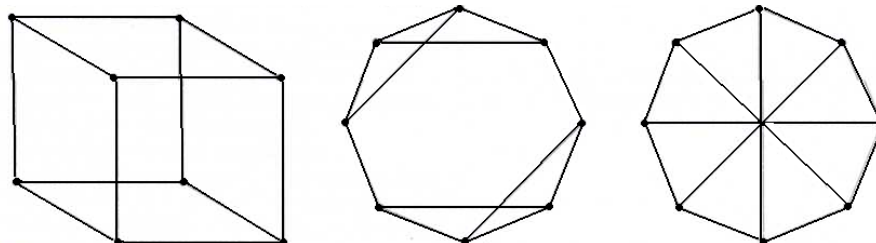
С помощью ПШ можно решить следующие задачи :

- 1) разбить множество вершин графа на его области связности;
- 2) для не совпадающих вершин $v_i, v_j \in V$ найти кратчайшую (v_i, v_j) - цепь;
- 3) в орграфе найти множество вершин, достижимых из заданной вершины v_i .

Трудоёмкость алгоритма поиска в ширину составляет $O(n^2)$. Известно, что трудоёмкость алгоритмов нахождения метрических характеристик составляет $O(n^3)$ операций.

Задачи и упражнения

1. Будут ли двудольными следующие графы



4. Будет ли двудольным цикл длины n (ответ зависит от n)?
5. Докажите, что граф, в котором есть цикл нечетной длины не может быть двудольным.
6. И обратно, если в графе нет циклов четной длины, то он двудольный.
7. Придумайте алгоритм проверки двудольности графа, просматривающим каждое его ребро только один раз.
8. В приведенном ниже графе найти двудольный подграф, содержащий максимальное число ребер. Доказать, что количество ребер есть максимально возможное, и выяснить является ли найденный подграф единственным с максимальным числом ребер.
9. Программирование алгоритма поиска в ширину (в глубину).
10. Составить программу нахождения метрических характеристик графа.
11. Составить программу нахождения компонент связности графа.
12. Составить программу нахождения множества вершин орграфа, достижимых из заданной вершины.

4. ЭЙЛЕРОВЫ ГРАФЫ

Цикл в графе $G=(V,E)$ называется эйлеровым, если он содержит все ребра графа, причем каждое ребро один раз. Связный граф $G=(V,E)$, в котором есть эйлеров цикл, называется эйлеровым.

Такой граф можно нарисовать, не отрывая карандаша от бумаги, и не повторяя линий.

Теорема 4.1. (Эйлер). Связный граф G является эйлеровым тогда и только тогда, когда степени всех его вершин четны.

Доказательство.

1. \Rightarrow Необходимость. Пусть G -эйлеров граф. Эйлеров цикл этого графа, проходя через каждую его вершину, входит в нее по одному ребру и выходит по другому. Это означает, что каждая вершина графа инцидентна четному числу ребер эйлерова цикла, а поскольку эйлеров цикл содержит все ребра графа G , то отсюда следует четность степеней всех его вершин.

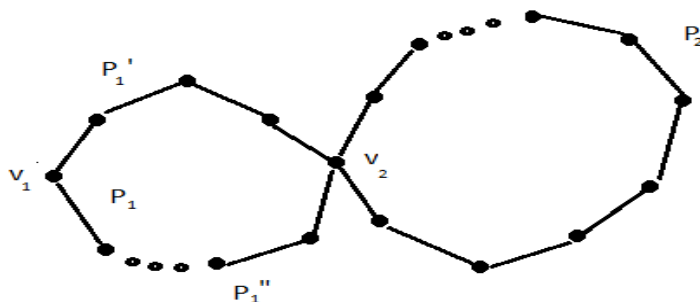


Рис.4.1. К доказательству теоремы Эйлера.

2. \Leftarrow Достаточность. Предположим, что степени вершины графа четны. Начнем строить цепь из произвольной вершины v_1 и будем продолжать ее насколько возможно, выбирая каждый раз новое ребро. Так как степени всех вершин графа G четны, то попав в очередную отличную от v_1 вершину, мы всегда будем иметь в распоряжении еще одно ребро. Поэтому цепь P_1 можно продолжить добавлением этого ребра. Таким образом построение цепи P_1 закончится в вершине $v_1 \in P_1$, т. е. P_1 -цикл. Если P_1 содержит все ребра графа G , то P_1 -эйлеров.

В противном случае удаляем все ребра цикла P_1 из G , получим G_1 . Поскольку G и P_1 имели вершины только четных степеней, то все вершины графа G_1 имеют четные степени. Кроме того, в силу связности графа G графы G_1 и P_1 должны иметь хотябы одну общую вершину v_2 . Начиная с вершины v_2 аналогичным образом построим цикл P_2 в графе G_1 . Обозначим через P_1' и P_1'' - части цикла P_1 от v_1 до v_2 и от v_2 до v_1 соответственно.

Получим новый цикл $P_3 = P'_1 \cup P_2 \cup P''_1$, который начиная с v_1 , проходит все ребра цепи P'_1 до v_2 , обходит все ребра цикла P_2 и возвращается в v_2 по ребрам цепи P''_1 . Если цикл P_1 - не эйлеров, проделав аналогичные построения, получим еще больший цикл. Этот процесс закончится построением эйлерова цикла. Теорема 4.1 доказана.

Возникает вопрос: как найти хотя бы один эйлеров цикл в эйлеровом графе G_1 , т.е. как перенумеровать все ребра графа числами $1, 2, 3, \dots$ чтобы номер, присвоенный ребру, указывал, каким по счету это ребро проходится в эйлеровом цикле? Это можно сделать, если применить следующий процесс:

1. Начиная с произвольной вершины u присваиваем произвольному ребру uv номер 1. Затем ребро и переходим в вершину v .

2. Пусть w – вершина, в которую мы пришли в результате выполнения предыдущего шага k , где k - номер, присвоенный некоторому ребру на этом шаге. Выбираем любое ребро инцидентное вершине w , причем мост выбираем в случае, если нет других ребер; присваиваем выбранному ребру номер $k + 1$ и вычеркиваем его.

Этот процесс, называемый алгоритмом Флёрри, заканчивается, когда все ребра окажутся вычеркнутыми, т.е. занумерованы. Трудоёмкость алгоритма Флёрри составляет $O(n^2)$ операций.

Теорема 4.2. Почти нет эйлеровых графов.

Задачи и упражнения

1. Существует ли эйлеров и гамильтонов циклы в графах из задачи предыдущего раздела?
2. Группа островов соединена мостами так, что от каждого острова можно добраться до любого другого. Турист обошел все острова, пройдя по каждому мосту ровно один раз. На острове Ташлы он побывал трижды. Сколько мостов ведет с Ташлы, если турист
 - а) не с него начал и не на нем закончил;
 - б) с него начал, но не на нем закончил;
 - с) с него начал и на нем закончил?
3. Доказать или опровергнуть следующие утверждения:
 - а) каждый эйлеров двудольный граф имеет четное число ребер;
 - б) каждый эйлеров граф с четным числом вершин имеет четное число ребер.
4. Составить программу нахождения Эйлерова цикла в графе.

5. ГАМИЛЬТОНОВЫ ГРАФЫ

Граф $G=(V,E)$ называется гамильтоновым, если он содержит простой цикл, содержащий каждую вершину. Этот цикл называется гамильтоновым. Во многих прикладных задачах требуется построить гамильтонову цепь. Игру «Кругосветное путешествие» предложил Гамильтон (рис.5.1).

Граф содержащий гамильтонову цепь называется трассируемым.

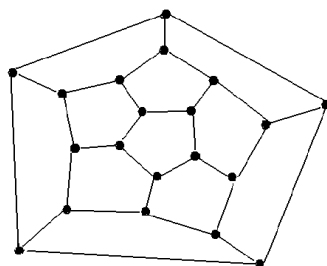


Рис. 5.1. Гамильтонов граф (додекаэдр).

Задача нахождения гамильтонова цикла в графе является NP-трудной задачей т.е. для её решения не существует полиномиальных алгоритмов.

Приведем некоторые достаточные и необходимые условия существования гамильтонова цикла в графе, касающиеся некоторых частных случаев.

Теорема 5.1. Простой граф порядка n произвольно гамильтонов тогда и только тогда, когда он является циклом, либо полным графом K_n , либо полным двудольным графом $K_{n/2,n/2}$. Последнее возможно, когда n - четное число.

Теорема 5.2. (В.Хватал). Граф G со степенной последовательностью d_1, d_2, \dots, d_n является гамильтоновым, если $\forall k: 1 \leq k \leq n/2$ истинна импликация $d_k \leq k \Rightarrow d_{n-k} \geq n-k$.

Теорема 5.3. (О.Оре). Если $\forall u, v \in V: u \neq v, |V| > 3$ графа $G=(V, E)$ выполняется неравенство $\deg u + \deg v \geq n$, то $G=(V, E)$ -гамильтонов.

Теорема 5.4. (Г.Дирак). Если $\forall v \in V: |V| > 3$ графа $G=(V, E)$ выполняется неравенство $\deg v > n/2$, то G -гамильтонов.

Теорема 5.5. (В.Перепелица). Почти все графы гамильтоновы.

Рассмотрим задачу Коммивояжера: коммивояжеру необходимо посетить несколько городов. Какой маршрут он должен избрать, чтобы начав двигаться из родного города, побывав в каждом

городе по одному разу, и затем вернуться домой по наикратчайшему пути?

Представим города в виде вершин, а дороги в виде ребер неориентированного взвешенного графа, длину дороги – весом соответствующего ребра. Если между каждой парой городов существует дорога, то задача коммивояжера сводится к отысканию наикратчайшего гамильтонова цикла в полном графе.

В полном графе порядка n существует $(n-1)!/2$ гамильтоновых циклов. Объем затрат даже для ЭВМ уже велик при $n \leq 50$. Для произвольного n полиномиального алгоритма для решения этой проблемы не существует.

Задачи и упражнения

1. Приведите пример негамильтонова графа, степени вершин которого равны (3,3,4,4,4,4). Степени вершин n -вершинного графа не меньше $n/2$. Докажите, что граф - гамильтонов.
2. Рассмотрим граф на множестве перестановок n элементов. Ребро графа соединяет две перестановки тогда и только тогда, когда перестановки отличаются транспозицией (т.е. перестановкой двух соседних элементов). Показать, что в таком графе имеется гамильтонов цикл.
3. Существуют ли графы, в которых эйлеров путь является также гамильтоновым путем? Если да, то охарактеризуйте эти графы, если нет — объясните почему.
4. Пусть простой граф $G = (V, E)$ задан следующим образом. Множество вершин графа есть множество $V = \{1, 2, \dots, 20\}$. Множество ребер графа есть $E = \{(a, b) / a, b \in G, a + b \equiv 2 \pmod{2}\}$.
 - а) Является ли граф G связным?
 - б) Является ли граф G двудольным?
 - в) Существует ли в графе G эйлеров путь (цикл)?
 - г) Существует ли в графе G гамильтонов цикл?
5. Пусть G -простой граф: $V = \{1, 2, \dots, 20\}$ - множество вершин, $E = \{(a, b) / a, b \in G, a + b \equiv 2 \pmod{2}\}$ - множество ребер.
 - а) Является ли граф G связным?
 - б) Является ли граф G двудольным?
 - в) Существует ли в графе G эйлеров путь (цикл)?
 - г) Существует ли в графе G гамильтонов цикл?
6. При каких m и n в графе K_{mn} существует а) эйлеров цикл; б) гамильтонов цикл? в) Постройте оба цикла для графа $K_{6,6}$.

6. ПЛОСКИЕ И ПЛАНАРНЫЕ ГРАФЫ

6.1. Определения

Плоским называется граф, вершины которого являются точками плоскости, а ребра - непрерывными плоскими линиями без самопересечений, соединяющими соответствующие вершины так, что никакие два ребра не имеют общих точек, кроме инцидентных им вершин.

Два графа G и G' называются изоморфными, если существует такое взаимно - однозначное соответствие вершин и ребер, что соответствующие ребра инцидентны соответствующим вершинам.

Любой граф, изоморфный плоскому графу, называется планарным.

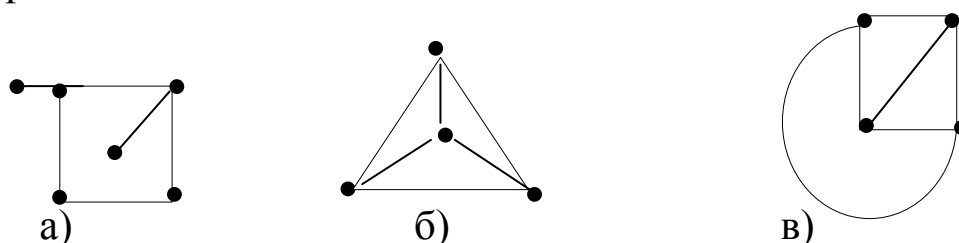


Рис.6.1. Планарные графы.

Граф K_4 является планарным, т.к. он изоморфен графу в)

рис.1. Очевидны следующие утверждения:

- 1) всякий подграф планарного графа планарен;
- 2) граф планарен тогда и только тогда, когда каждая компонента связности - планарный граф.

Головоломка «три дома, три колодца» наводит на мысль, что существуют не только планарные графы.

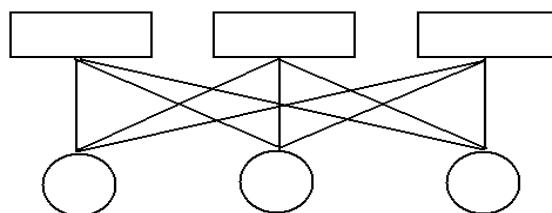


Рис.6.2. Схема головоломки «три дома, три колодца».

Восемь дорожек не пересекаются, а девятую невозможно нарисовать так, чтобы она не пересекалась с одной из восьми.

6.2. Грани плоского графа. Формула Эйлера

Гранью плоского называется максимальное по включению множество точек плоскости, каждая пара которых может быть соединена жордановой кривой, не пересекающей ребра графа. Причем, каждая точка плоскости принадлежит хотя бы одной грани плоского графа. Границей грани будем называть множество вершин и ребер, принадлежащих этой грани. Отметим, что одна из граней плоского графа является внешней.

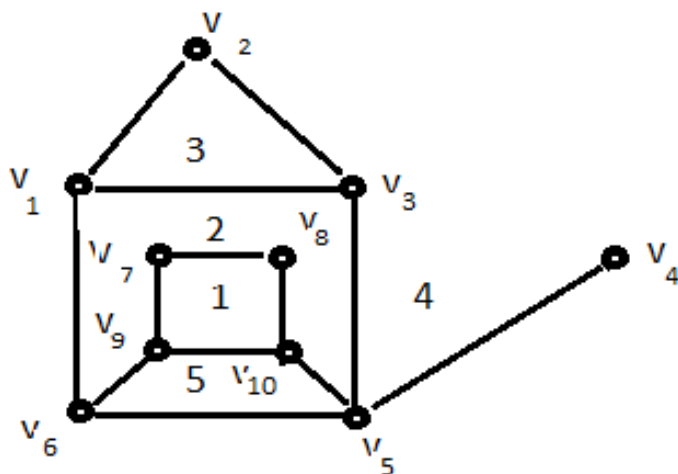


Рис. 6.3. Плоский граф.

Пример 6.1. Грань 4 на рис. 6.3 плоского графа – внешняя.

Рассмотрим свойства планарных графов.

Свойство 1. *Всякий планарный граф допускает такую плоскую укладку, в которой всякая вершина или ребро графа будет принадлежать некоторой внешней грани.*

Свойство 2. *Пусть граф G состоит из двух компонент связности G_1 и G_2 , являющихся плоскими графами и произвольно выбраны вершины $v_1 \in V_1$ и $v_2 \in V_2$. Тогда граф G , полученный слиянием вершин v_1, v_2 в вершину v , имеет плоскую укладку. При этом вершина v – точка сочленения графа G .*

Свойство 3. *Любые две вершины, принадлежащие границе некоторой грани плоского графа, можно соединить простой цепью произвольной длины так, что грань разобьётся на две.*

Свойство 4. *Для любого плоского графа каждая точка плоскости, не лежащая на ребре, входит только в одну грань, а каждая точка ребра, не являющаяся вершиной, входит только в одну грань, если это ребро-мост, и точно в две грани, если это ребро-не мост.*

Теорема 6.1 (Эйлера). Для всякого связного плоского графа верно равенство:

$$n - m + f = 2. \quad (6.1)$$

Доказательство. Пусть T - некоторый остов графа G , тогда $m = n - 1$, $f = 1$. Так как T - дерево, то $n - m + f = n - (n - 1) + 1 = 2$ и равенство (6.1) верно. Будем добавлять к остову T поочерёдно все недостающие ребра графа G . Тогда на каждом шаге такой процедуры согласно свойству 3 число рёбер m и число граней f увеличиваются на единицу: $m := m + 1$, $f := f + 1$, а число вершин n остается постоянным. Следовательно, формула (6.1) справедлива для любого связного плоского графа. Теорема 6.1 доказана.

Следствие 6.1. Для всякого выпуклого многогранника сумма числа вершин n и числа граней f без числа рёбер равна 2, т.е. $n - m + f = 2$.

Следствие 6.2. Для связного планарного (n, m) - графа верно неравенство: $m \leq 3n - 6$ при $n > 3$.

Теорема 6.2. Граф K_5 не планарен.

Доказательство. Предположим, что граф K_5 - планарен. Поскольку K_5 имеет $n = 5$ вершин и $m = 10$ рёбер, то по следствию 6.2 должно вытекать неравенство $m \leq 3n - 6$, тогда $10 \leq 9$ что неверно.

Теорема 6.3. Граф $K_{3,3}$ не планарен.

Доказательство. Пусть $K_{3,3}$ - планарный, тогда поскольку он связный должно выполняться неравенство $m \leq 3n - 6$; тогда $n = 6$, $m = 9$. Для его плоской укладки выполнялось бы:

$n - m + f = 6 - 9 + f \Rightarrow f = 5$. Тогда всякая грань двудольного графа $K_{3,3}$ должна быть ограничена, по крайней мере, четырьмя ребрами. Отсюда справедливо неравенство $4f \leq 2m \Rightarrow 20 \leq 18$. Следовательно, получили противоречие.

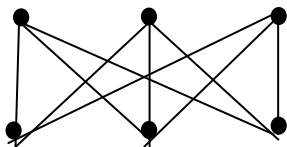


Рис. 6.4. Не планарный граф $K_{3,3}$.

Теорема 6.4. Связный граф планарен тогда и только тогда, когда каждый его блок планарен.

6.3. Критерий планарности графа

Рассмотрим операцию подразделения ребра. Эта операция состоит в том, что ребро $e=(a,b)$ удаляется, и добавляются два новых ребра $e_1=(a,v), e_2=(v,b)$ где v - новая вершина.

Два графа называются гомеоморфными, если они могут быть получены из одного и того же графа подразбиением его рёбер.

Теорема 6.5. (Понтрягина – Куратовского).

Граф планарен тогда и только тогда, когда он не содержит подграфов, гомеоморфных K_5 и $K_{3,3}$.

Задачи и упражнения

В этом параграфе буквами n, m, f обозначается число вершин, ребер и граней плоского графа.

1. Если $f=0$, то G не содержит циклов (а) почему?), а поскольку по условию G связан, то G - дерево.
2. При $f=0$ теорема утверждает, что $n-m=1$, что для деревьев верно (б) почему?).
3. Если $f>0$, то в графе существует цикл, ограничивающий одну из граней. Сотрем одно из ребер этого цикла. В новом графе вершин останется столько же, а ребер и граней станет на одну меньше (в) почему?). Поэтому значение выражения $1-n+f$ не изменится.
4. г) Докажите, что граф останется связным. Будем повторять эту операцию, пока не кончатся грани. Выше доказано, что получившийся граф будет деревом, т.е. для него $n-m+f=n-m=1$. Но в процессе стирания ребер число $n-m+f$ не менялось, поэтому $n-m+f=1$ и у исходного графа.
5. Объясните, почему число граней плоского графа не зависит от способа, которым он нарисован на плоскости.
6. В плоском графе имеется 7 граней, 3 треугольника, 3 четырехугольника и 5-угольник. Сколько в этом графе ребер? А вершин?
7. Докажите, что для планарного связного графа справедливы

неравенства

а) $2m > 3f$;

б) $m \leq 3n - 6$.

8. Докажите, что граф K_5 не планарен.
9. Докажите, что граф, имеющий 10 вершин, степень каждой из которых равна 5, — не планарен.
10. Приведите пример планарного графа, степень каждой из вершин которого равна 5.
11. Докажите, что в плоском графе есть вершина, степень которой не превосходит 5.
12. Каждое ребро полного графа с 11 вершинами покрашено в один из двух цветов: красный или синий. Докажите, что либо "красный", либо "синий" граф не планарен.
13. Пусть все ребра плоского графа - не менее, чем 4-угольники. Докажите, что для такого графа $m > 2f$.
14. Выведите из предыдущей задачи, что граф $K_{3,3}$ не планарен.
15. При каких m и n будут планарны графы:
 - а) K_n ?
 - б) $K_{m,n}$?
 - в) $2n$ -угольник с проведенными главными диагоналями?
- 16*. Будет ли планарен каркас 4-мерного гиперкуба?

7. НАХОЖДЕНИЕ КРАТЧАЙШИХ ПУТЕЙ В ГРАФАХ

7.1. Постановка задачи

Пусть дан граф $G = (X, \Gamma)$, дугам которого приписаны веса (стоимости), задаваемые матрицей $C = [c_{ij}]$. Задача о кратчайшем пути состоит в нахождении кратчайшего пути от заданной начальной вершины $s \in X$ до заданной конечной вершины $t \in X$, при условии, что такой путь существует, т.е. при условии $t \in R(s)$. Здесь $R(s)$ - множество, достижимое из вершины s . Элементы c_{ij} матрицы весов C могут быть положительными, отрицательными или нулями. Единственное ограничение состоит в том, чтобы в G не было циклов с отрицательным суммарным весом. Если такой цикл все же существует и x_i - некоторая его вершина, то, двигаясь от s к x_i , обходя затем цикл достаточно большое число раз и попадая, наконец, в t , мы получим путь со сколь угодно малым весом, означаящим, что кратчайшего пути не существует.

Следующие задачи являются обобщениями сформулированной выше задачи о кратчайшем пути.

1. Для заданной начальной вершины s найти кратчайшие пути между s и всеми другими вершинами $x_i \in X$.

2. Найти кратчайшие пути между всеми парами вершин.

На практике часто требуется найти не только кратчайший путь, но также второй, третий и т.д. кратчайшие пути в графе. Располагая этими результатами, можно решить, какой путь выбрать в качестве наилучшего (указанный подход полезен при использовании таких критериев, которые являются субъективными по своей природе или не могут быть непосредственно включены в алгоритм).

7.2. Алгоритм Дейкстры

Наиболее эффективный алгоритм решения задачи о кратчайшем $(s-t)$ -пути первоначально дал Дейкстра. Этот метод основан на приписывании вершинам временных меток, причем метка вершины дает верхнюю границу длины пути от s к этой вершине. Значения меток постепенно уменьшаются с помощью некоторой итерационной процедуры, и на каждом шаге итерации одна из временных меток становится постоянной. Последнее указывает на то, что

метка уже не является верхней границей, а дает точную длину кратчайшего пути от s к рассматриваемой вершине.

Описание алгоритма Дейкстра.

Пусть $l(x_i)$ - метка вершины x_i .

Шаг 1. Положить $l(s) = 0$ и считать эту метку постоянной. Положить $l(x_i) = \infty$ для всех $x_i \neq s$ и считать эти метки временными. Положить $p = s$.

Шаг 2. Для всех $x_i \in \Gamma(p)$, метки которых временные, изменить метки в соответствии со следующим выражением:

$$l(x_i) = \min [l(x_i), l(p) + c(p, x_i)] \quad (7.1)$$

Шаг 3. Среди всех вершин с временными метками найти такую, для которой $l(x_i^*) = \min [l(x_i)]$.

Шаг 4. Считать метку вершины x_i^* постоянной и положить $p = x_i^*$.

Шаг 5. 1) Если $p = t$, то $l(p)$ является длиной кратчайшего пути. Если $p \neq t$, перейти к шагу 2 (для случая поиска пути от s к t .)

2) Если все вершины помечены как постоянные, то эти метки дают длины кратчайших путей. Если некоторые метки являются временными перейти к шагу 2 (для случая поиска путей от s ко всем остальным вершинам).

Как только длины кратчайших путей от s будут найдены, то сами пути можно получить при помощи рекурсивной процедуры с использованием соотношения (7.2). Так как вершина x'_i непосредственно предшествует вершине x_i в кратчайшем пути от s к x_i , то для любой вершины x_i соответствующую вершину x'_i можно найти как одну из оставшихся вершин, для которой

$$l(x'_i) + c(x'_i, x_i) = l(x_i). \quad (7.2)$$

Если кратчайший путь от s до любой вершины x_i является единственным, то дуги (x'_i, x_i) этого кратчайшего пути образуют ориентированное дерево с корнем s . Если существует несколько "кратчайших" путей от s к какой-либо другой вершине, то при некоторой фиксированной вершине x'_i соотношение (7.2) будет выполняться для более чем одной вершины x_i . В этом случае выбор может быть либо произвольным (если нужен какой-то один кратчайший путь между s и x_i), либо таким, что рассматриваются все

дуги (x'_i, x_i) , входящие в какой-либо из кратчайших путей, и при этом совокупность всех таких дуг образует не ориентированное дерево, а общий граф, называемый базой относительно s . Трудоемкость алгоритма Дейкстра составляет $O(n^3)$ операций.

Пример 7.1. Рассмотрим граф G , изображенный на рис. 7.1, где каждое неориентированное ребро рассматривается как пара противоположно ориентированных дуг равного веса. Матрица весов C приведена ниже.

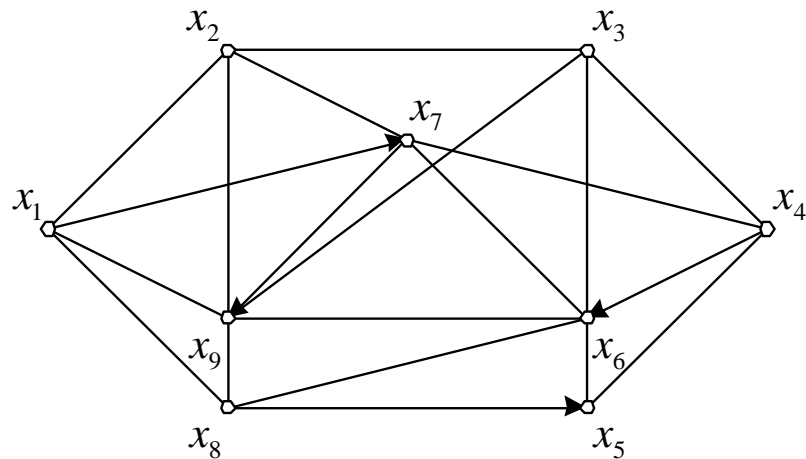


Рис. 7.1. Граф G .

Требуется найти все кратчайшие пути от вершины x_1 ко всем остальным вершинам. Постоянные метки будем снабжать знаком $+$, остальные метки рассматриваются как временные.

$$C = \begin{pmatrix} & x_1 & x_2 & x_3 & x_4 & x_5 & x_6 & x_7 & x_8 & x_9 \\ x_1 & 0 & 12 & 0 & 0 & 0 & 0 & 5 & 2 & 10 \\ x_2 & 12 & 0 & 14 & 0 & 0 & 0 & 4 & 0 & 15 \\ x_3 & 0 & 14 & 0 & 20 & 0 & 25 & 0 & 0 & 9 \\ x_4 & 0 & 0 & 20 & 0 & 2 & 15 & 5 & 0 & 0 \\ x_5 & 0 & 0 & 0 & 2 & 0 & 8 & 0 & 0 & 0 \\ x_6 & 0 & 0 & 25 & 0 & 8 & 0 & 12 & 11 & 10 \\ x_7 & 0 & 4 & 0 & 5 & 0 & 12 & 0 & 0 & 22 \\ x_8 & 2 & 0 & 0 & 0 & 24 & 11 & 0 & 0 & 7 \\ x_9 & 10 & 15 & 0 & 0 & 0 & 10 & 22 & 7 & 0 \end{pmatrix}$$

Воспользуемся алгоритмом Дейкстра.

Шаг 1. $l(x_1) = 0^+$, $l(x_i) = \infty \forall x_i \neq x_1$, $p = x_1$.

Первая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_1) = \{x_2, x_7, x_8, x_9\}$ - все метки временные.

Возьмем сначала x_2 . Из (7.1) получаем $l(x_2) = \min[\infty, 0^+ + 12] = 12$, аналогично $l(x_7) = 5$, $l(x_8) = 2$, $l(x_9) = 10$.

Шаг 3. $\min \begin{bmatrix} 12, 5, 2, 10, & \infty \\ x_2, x_7, x_8, x_9, & x_3, x_4, x_5, x_6 \end{bmatrix} = 2$ соответствует x_8 .

Шаг 4. x_8 получает постоянную метку $l(x_8) = 2^+$, $p = x_8$.

Шаг 5. Не все вершины имеют постоянные метки, поэтому переходим к шагу 2. Метки в начале следующей итерации показаны на рис. 7.2(а).

Вторая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_8) = \{x_5, x_6, x_9\}$ - все метки временные. Из со-

отношения (1.1) имеем $l(x_5) = \min[\infty, 2^+ + 24] = 26$,

аналогично $l(x_6) = 13$, $l(x_9) = 9$. Метки изображены на рис. 7.2(б).

Шаг 3. $\min \begin{bmatrix} 12, 26, 13, 5, 9, & \infty \\ x_2, x_5, x_6, x_7, x_9, & x_3, x_4 \end{bmatrix} = 5$ соответствует x_7 .

Шаг 4. x_7 получает постоянную метку $l(x_7) = 5^+$, $p = x_7$.

Шаг 5. Перейти к шагу 2.

Третья итерация

Шаг 2. $\Gamma(p) = \Gamma(x_7) = \{x_2, x_4, x_6, x_9\}$ - из соотношения (7.1) по-

лучаем $l(x_2) = \min[12, 5^+ + 4] = 9$,

и аналогично $l(x_4) = 10$, $l(x_6) = 13$, $l(x_9) = 9$.

Шаг 3. $\min \begin{bmatrix} 9, 10, 26, 13, 9, & \infty \\ x_2, x_4, x_5, x_6, x_9, & x_3 \end{bmatrix} = 9$ соответствует x_2 .

Шаг 4. x_2 получает постоянную метку $l(x_2) = 9^+$, $p = x_2$.

Шаг 5. Перейти к шагу 2.

Четвертая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_2) = \{x_3, x_7, x_9\}$ - не все метки временные, по-

этому из соотношения (7.1) получаем $l(x_3) = \min[\infty, 9^+ + 14] = 23$, и

аналогично $l(x_9) = 9$.

Шаг 3. $\min \begin{bmatrix} 23, 10, 26, 13, 9 \\ x_3, x_4, x_5, x_6, x_9 \end{bmatrix} = 9$ соответствует x_9 .

Шаг 4. x_9 получает постоянную метку $l(x_9) = 9^+$, $p = x_9$.

Шаг 5. Перейти к шагу 2.

Пятая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_9) = \{x_2, x_6, x_7, x_8\}$ - не все метки временные, поэтому из соотношения (7.1) получаем $l(x_6) = \min[13, 9^+ + 10] = 13$.

Шаг 3. $\min \begin{bmatrix} 23, 10, 26, 13 \\ x_3 \quad x_4 \quad x_5 \quad x_6 \end{bmatrix} = 10$ соответствует x_4 .

Шаг 4. x_4 получает постоянную метку $l(x_4) = 10^+$, $p = x_4$.

Шаг 5. Перейти к шагу 2.

Шестая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_4) = \{x_3, x_5, x_6\}$ - все метки временные, из соотношения (1.1) получаем $l(x_3) = \min[23, 10^+ + 20] = 23$, и аналогично $l(x_5) = 12$, $l(x_6) = 13$.

Шаг 3. $\min \begin{bmatrix} 23, 12, 13 \\ x_3 \quad x_5 \quad x_6 \end{bmatrix} = 12$ соответствует x_5 .

Шаг 4. x_5 получает постоянную метку $l(x_5) = 12^+$, $p = x_5$.

Шаг 5. Перейти к шагу 2.

Седьмая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_5) = \{x_4, x_6\}$ - не все метки временные, поэтому из соотношения (7.1) получаем

$l(x_6) = \min[13, 12^+ + 8] = 13$.

Шаг 3. $\min \begin{bmatrix} 23, 13 \\ x_3 \quad x_6 \end{bmatrix} = 13$ соответствует x_6 .

Шаг 4. x_6 получает постоянную метку $l(x_6) = 13^+$, $p = x_6$.

Шаг 5. Перейти к шагу 2.

Восьмая итерация

Шаг 2. $\Gamma(p) = \Gamma(x_6) = \{x_3, x_5, x_8, x_9\}$ - не все метки временные, поэтому из (7.1) получаем: $l(x_3) = \min[23, 13^+ + 25] = 23$.

Шаг 3. $\min \begin{bmatrix} 23 \\ x_3 \end{bmatrix} = 23$ соответствует x_3 .

Шаг 4. x_3 получает постоянную метку $l(x_3) = 23^+$, $p = x_3$.

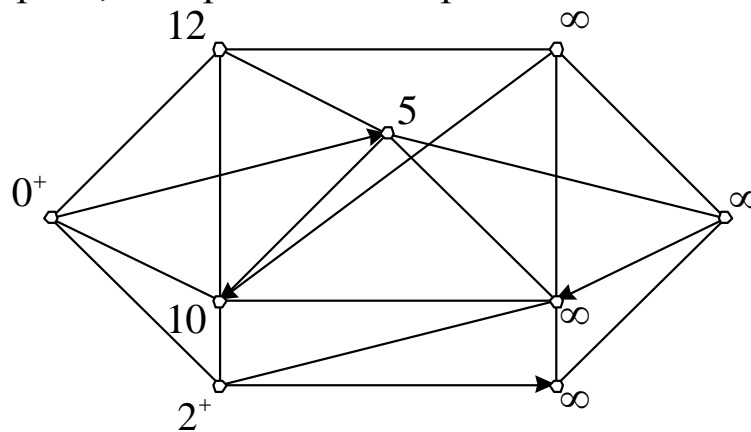
Шаг 5. Все вершины имеют постоянные метки. Конец работы алгоритма. Метки, полученные в результате работы алгоритма, показаны на рис. 7.2(в).

Найдем кратчайший путь между вершиной x_2 и начальной вершиной x_1 , последовательно используя соотношение (7.2). Таким образом, полагая $x_i = x_2$, находим вершину x'_2 , непосредственно предшествующую x_2 в кратчайшем пути от x_1 к x_2 : вершина x'_2 должна удовлетворять соотношению $l(x'_2) + c(x'_2, x_2) = l(x_2) = 9$.

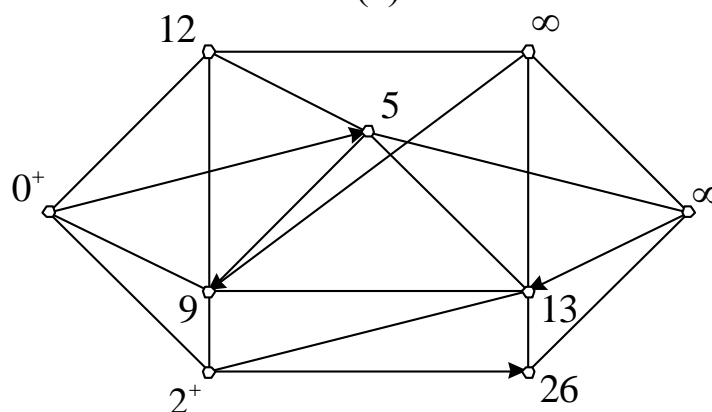
Единственной такой вершиной является x_7 . Далее, применяем второй раз соотношение (7.2), беря $x_i = x_7$; получаем вершину x'_7 , непосредственно предшествующую x_7 в кратчайшем пути от x_1 к x_2 . Вершина x'_7 удовлетворяет соотношению $l(x'_7) + c(x'_7, x_7) = l(x_7) = 5$.

Единственной такой вершиной является x_1 и поэтому кратчайший путь от x_1 к x_2 , есть (x_1, x_7, x_2) .

Пусть x_1 - база, дающая все кратчайшие пути от x_1 , представляет собой дерево, изображенное жирными линиями на рис. 7.2 (в).



(a)



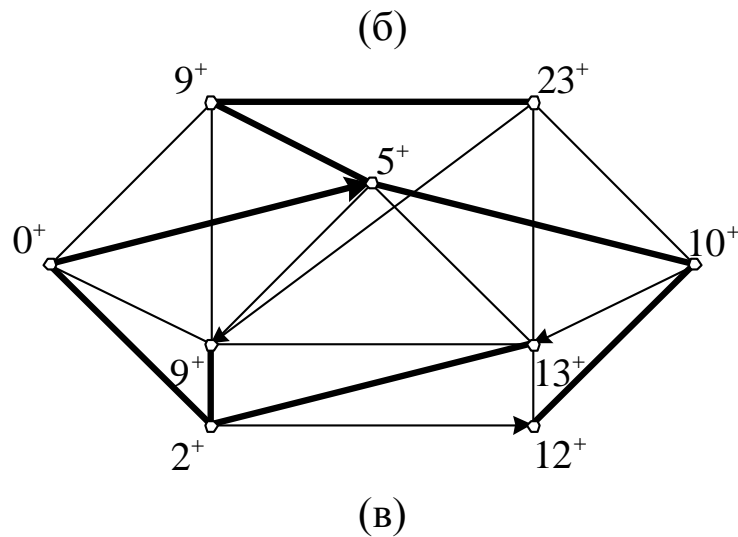


Рис. 7.2. (а) Метки в конце 1-й итерации.
 (б) Метки в конце шага 2 на 2-й итерации.
 (в) Окончательные метки вершин и x_1 - база.

7.3. Алгоритм Форда (случай общей матрицы весов)

Алгоритм Дейкстра применим лишь в том случае, когда $c_{ij} \geq 0$ для всех i и j . Однако если матрица C является матрицей стоимостей, то дуги, приносящие доход, должны иметь отрицательные «стоимости». В этом случае для нахождения кратчайших путей между вершиной s и всеми другими вершинами можно воспользоваться описанной ниже процедурой. Этот метод также является итерационным и основан на метках вершин, причем в конце k -й итерации метки равны длинам тех кратчайших путей (от s ко всем остальным вершинам), которые содержат не более $k+1$ дуг. В отличие от алгоритма Дейкстра, никакая из меток во время этого процесса не рассматривается как окончательная. Описываемый метод был первоначально предложен Фордом.

Алгоритм Форда

Пусть $l^k(x_i)$ - метка вершины x_i в конце $(k+1)$ -й итерации.

Шаг 1. Положить $S = \Gamma(s)$, $k=1$, $l^1(s) = 0$, $l^1(x_i) = c(s, x_i)$ для всех $x_i \in \Gamma(s)$ и $l^1(x_i) = \infty$ для всех остальных x_i .

Шаг 2. Для каждой вершины $x_i \in \Gamma(s)$ ($x_i \neq s$) изменить ее метку следующим образом:

$$l^{k+1}(x_i) = \min \left[l^k(x_i), \min_{x_j \in T_i} \{ l^k(x_j) + c(x_j, x_i) \} \right],$$

где $T_i = \Gamma^{-1}(x_i) \cap S$ (множество T_i содержит те вершины, для которых текущие кратчайшие пути из s состоят из k дуг и для которых существуют дуги к вершине x_i). Для вершин $x_i \notin \Gamma(s)$ положим $l^{k+1}(x_i) = l^k(x_i)$.

Шаг 3.

1. Если $k \leq n-1$ и $l^{k+1}(x_i) = l^k(x_i)$ для всех x_i , то получен оптимальный ответ и метки равны длинам кратчайших путей.

2. Если $k < n-1$ и $l^{k+1}(x_i) \neq l^k(x_i)$ для некоторой вершины x_i , то перейти к шагу 4.

3. Если $k = n-1$ и $l^{k+1}(x_i) \neq l^k(x_i)$ для некоторой вершины x_i , то в графе существует цикл отрицательного веса и задача не имеет решения.

Шаг 4. Обновить множество S следующим образом:

$$S = \{x_i \mid l^{k+1}(x_i) \neq l^k(x_i)\}.$$

Шаг 4. Положить $k = k + 1$ и перейти к шагу 2.

Также можно привести общую матрицу весов к неотрицательной, увеличив вес каждого ребра на величину $\max[abs(c_{ij})]$ и найти кратчайшие пути между вершиной s и всеми другими вершинами с помощью алгоритма Дейкстры.

Трудоёмкость алгоритма Форда составляет $O(n^3)$ операций.

7.4. Алгоритм Флойда нахождения кратчайших путей между всеми парами вершин

Пусть требуется найти кратчайшие пути между всеми парами вершин графа. Очевидный способ получить ответ на этот вопрос заключается в n -кратном применении алгоритма Дейкстры или алгоритма Форда, причем каждый раз в качестве начальной вершины s берутся различные вершины. В случае полного графа с неотрицательной матрицей весов C сложность алгоритма Форда $O(n^3)$, а для произвольной матрицы весов - $O(n^4)$.

Опишем совершенно иной подход к задаче нахождения кратчайших путей между всеми парами вершин. Этот метод применим как к неотрицательным, так и к произвольным матрицам весов и время, необходимое для вычислений, пропорционально n^3 . Если этот метод применим к графам с неотрицательной матрицей весов,

то он сэкономит почти 50 % времени по сравнению с n -кратным применением алгоритма Дейкстры. Метод был предложен первоначально Флойдом. Он базируется на использовании последовательности из n преобразований (итераций) начальной матрицы весов C . При этом на k -й итерации матрица представляет длины кратчайших путей между каждой парой вершин с тем ограничением, что путь между x_i и x_j (для любых x_i и x_j) содержит в качестве промежуточных только вершины из множества $\{x_1, x_2, \dots, x_k\}$.

Описание алгоритма Флойда:

Предположим, что в начальной матрице весов $c_{ii} = 0$ для всех $i = 1, 2, \dots, n$ и $c_{ij} = \infty$, если в графе отсутствует дуга (x_i, x_j) .

Шаг 1. Присвоить $k = 0$.

Шаг 2. $k = k + 1$.

Шаг 3. Для всех $i \neq k$, таких, что $c_{ik} \neq \infty$, и для всех $j \neq k$, таких, что $c_{kj} \neq \infty$, введем операцию

$$c_{ij} = \min[c_{ij}, c_{ik} + c_{kj}]. \quad (7.3)$$

Шаг 4.

1. Если $c_{ii} < 0$, то в графе G существует цикл отрицательного веса, содержащий вершину x_i , и решения нет.

2. Если все $c_{ii} \geq 0$ и $k = n$, то получено решение. Матрица $[c_{ij}]$ дает длины всех кратчайших путей.

3. Если все $c_{ii} \geq 0$, но $k < n$, то вернуться к шагу 2.

Сами кратчайшие пути можно найти по их длинам с помощью рекурсивной процедуры, подобной той, которая выше определялась соотношением (7.2). С другой стороны, можно использовать информацию о самих путях (наряду с информацией о длинах путей). Этот последний метод полезен в тех случаях, когда требуется найти в графе цикл отрицательного веса (если такой существует). В этом методе в дополнение к матрице весов C хранится и обновляется вторая $(n \times n)$ -матрица $\Theta = [\theta_{ij}]$. Элемент θ_{ij} указывает вершину, непосредственно предшествующую вершине x_j в кратчайшем пути от x_i к x_j . Матрице Θ присваиваются начальные значения $\theta_{ij} = x_i$ для всех x_i и x_j . В соответствии с (7.3) на шаге 3 алгоритма обновление матрицы происходит так:

$$\theta_{ij} = \begin{cases} \theta_{kj}, & \text{если } c_{ik} + c_{kj} < c_{ij} \\ \text{не изменяется,} & \text{если } c_{ik} + c_{kj} \geq c_{ij} \end{cases}.$$

В конце алгоритма кратчайшие пути получаются непосредственно из заключительной матрицы Θ . Таким образом, кратчайший путь между двумя вершинами x_i и x_j дается следующей последовательностью вершин:

$$x_i, x_\nu, \dots, x_\gamma, x_\beta, x_\alpha, x_j,$$

где $x_\alpha = \theta_{ij}$, $x_\beta = \theta_{i\alpha}$, $x_\gamma = \theta_{i\beta}$ и т.д. до $x_i = \theta_{i\nu}$.

Здесь следует отметить, что если всем c_{ii} придать начальные значения ∞ (а не 0), то конечное значение величины c_{ii} будет равно весу цепи, проходящей через вершину x_i . Исходя из структуры матрицы Θ , полученной в процессе той итерации, когда элемент c_{ii} становится отрицательным, можно найти цикл отрицательного веса, соответствующий этому элементу. Трудоемкость алгоритма Флойда составляет $O(n^4)$ операций.

7.5. Применение к сетевому планированию и управлению

Предположим, что нужно реализовать некий проект, состоящий из большого числа этапов. Каждому этапу поставим в соответствие вершину, и если этап i предшествует этапу j , то вершины соединяются дугой (x_i, x_j) , направленной от вершины x_i к вершине x_j . Каждой дуге приписывается некоторый вес c_{ij} , равный минимальной задержке во времени между началом этапа i и началом этапа j .

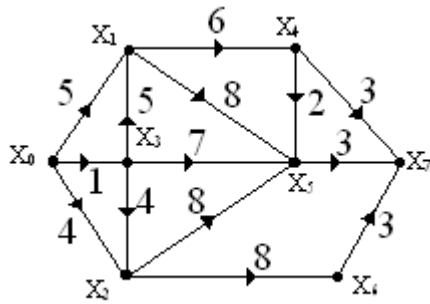
Требуется найти минимальное время, необходимое для реализации проекта. Иными словами, нужно найти в графе самый длинный путь между вершиной s , изображающей начало, и вершиной t , изображающей завершение всех необходимых для реализации проекта работ. Самый длинный путь называется критическим путем, так как этапы, относящиеся к этому пути, определяют полное время реализации проекта, и всякая задержка с началом выполнения любого из этих этапов приведет к задержке выполнения проекта в целом.

Данную задачу можно решить как задачу нахождения кратчайшего пути, используя алгоритм Дейкстры, заменив операцию \min на \max .

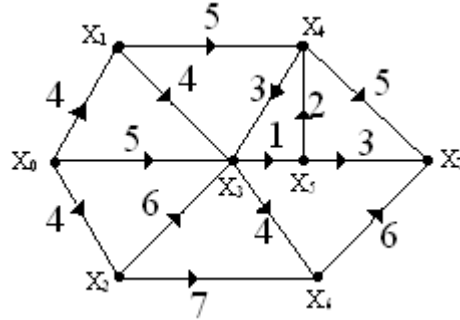
Задачи и упражнения

1. Найти путь минимальной длины (Рис.7.3 между x_0 и x_7 и его длину).

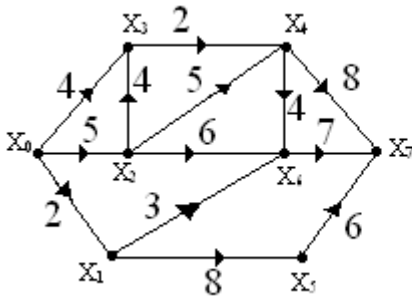
2. Задан сетевой граф проекта (Рис.7.3). Найти критический путь и минимальное время проекта



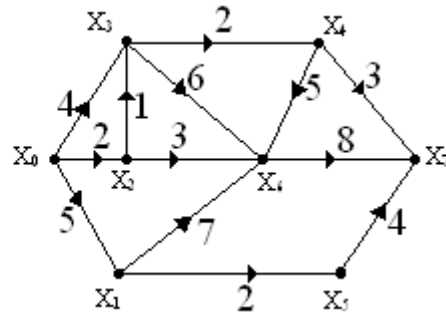
a)



b)



c)



d)

Рис. 7.3. Орграфы.

3. Составить программу нахождения кратчайшего пути между двумя вершинами в графе. (Программирование алгоритма Дейкстры).

8. ВЕРШИННАЯ РАСКРАСКА ГРАФА

Разнообразные задачи, возникающие при планировании производства, составлении графиков осмотра, хранения и транспортировке товаров и т.д., могут быть представлены часто как задачи теории графов, тесно связанные с так называемой «задачей раскраски». Графы, рассматриваемые в этой главе, являются простыми неориентированными.

8.1. Постановка задачи

Пусть задан граф $G=(V,E)$ с $n=|V|$ вершинами и некоторое натуральное число $k \leq |V|$. Произвольная функция вида $f:V \rightarrow \{1,2,\dots,k\}$ называется вершинной k -раскраской или просто k -раскраской графа G . Раскраска называется правильной, если $f(u) \neq f(v)$ для любых смежных вершин $u \neq v$ графа G . Граф, для которого существует правильная k -раскраска, называется k -раскрашиваемым.

Минимальное число k , при котором граф G является k -раскрашиваемым, называется хроматическим числом и обозначается $\chi(G)$. Если $\chi(G)=k$, то граф G называется k -хроматическим. Правильная k -раскраска графа G при $k=\chi(G)$ называется минимальной.

Гипотеза четырех красок формулируется следующим образом: всякий планарный граф 4-раскрашиваем. В 1976 году Хейкеном и Appelем было получено доказательство этой гипотезы. Отметим, что это чрезвычайно длинное и сложное доказательство трудно воспроизводимо, поскольку было проведено с помощью компьютера. Конечно, такое доказательство не удовлетворяет многих математиков. До сих пор нет уверенности в том, что эта гипотеза на самом деле доказана.

Лемма 8.1. Всякий планарный граф с $n \geq 4$ вершинами имеет по крайней мере 4 вершины со степенями, не превосходящими 5.

Теорема 8.1. (Р. Хивуд). Всякий планарный граф G 5-раскрашиваем (т.е. если G — планарный граф, то $\chi(G) \leq 5$).

Доказательство. Не теряя общности, будем рассматривать плоские графы. Проведем индукцию по числу вершин графа G .

Пусть G - плоский граф с $n=|V|$ вершинами. Известно, что при $n \leq 5$ граф G является 5-раскрашиваемым и теорема 8.1 справедлива.

Предположим, что теорема 8.1 верна для всех графов порядка, не превосходящего n , $n \geq 5$. Рассмотрим произвольный плоский граф порядка $n+1$. Согласно лемме 8.1. этот граф содержит вершину v_0 , степень которой не превосходит пяти. Рассмотрим возможные случаи.

1) $|N_{v_0}| \leq 4$. По индуктивному предположению граф $G' = G - v_0$

5-раскрашиваем; раскрасим его вершины пятью цветами. Затем окрасим вершину v_0 в тот из пяти цветов, который не был использован при раскраске вершин из N_{v_0} . Отсюда следует, что в мно-

жестве N_{v_0} имеются хотя бы две пары различных несмежных вер-

шин с различными соответствующими компонентами (например, на рис.8.1 это пары (v_3, v_5) и (v_2, v_4)).

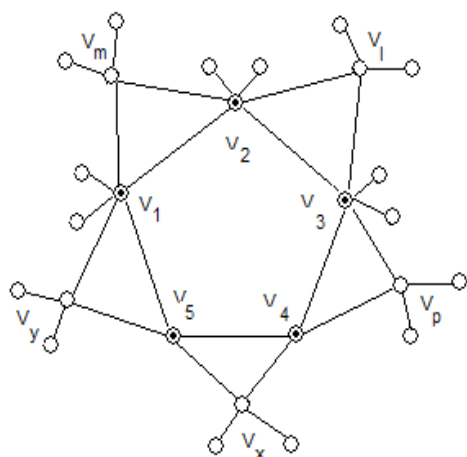


Рис.8.1. Граф $G' = G - v_0$.

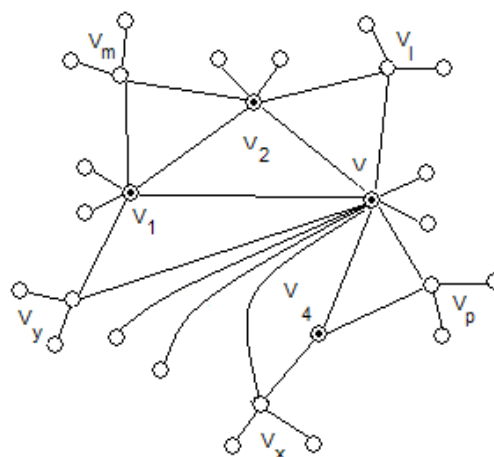


Рис.8.2. Граф G'' .

2) $|N_{v_0}| = 5$. В множестве N_{v_0} имеются две различные несмежные вершины v_1 и v_3 , иначе граф G содержит подграф, стягиваемый к K_5 , и поэтому не является планарным. Граф G'' , полученный из $G' = G - v_0$ слиянием этих вершин в вершину v , является плоским и по индуктивному предположению 5-раскрашиваемым. Фиксируем какую-либо из правильных 5-раскрасок графа G'' . В графе G окрасим вершины v_1 и v_3 в цвет вершины v , а остальные отличные от v_1 и v_3 вершины окрасим в те же цвета, что и соответ-

ствующие вершины графа G'' . Затем окрасим вершину v_0 в цвет, не использованный при раскраске вершин множества N_{v_0} .

Получена правильная 5 - раскраска графа G . Таким образом, теорема 8.1 доказана.

Заметим, что теоремы о 2 - раскраске деревьев, о k - раскраске планарных графов при $k > 4$ доказываются аналогично теореме 8.1. Известны следующие две теоремы.

Теорема 8. 2. Следующие два утверждения эквивалентны:

- 1) произвольный плоский граф 4- раскрашиваем;
- 2) любая кубическая карта 4- раскрашивается.

Теорема 8. 3. Следующие два утверждения эквивалентны:

- 1) гипотеза четырех красок верна;
- 2) любой связный 5-хроматический граф стягиваем к K_5 .

Соответствующая хроматическому числу раскраска вершин разбивает множество вершин графа на r подмножеств, каждое из которых содержит вершины одного цвета. Эти множества являются независимыми, поскольку в пределах одного множества нет двух смежных вершин.

Хроматическое число графа нельзя найти, зная только числа вершин и ребер графа. Недостаточно также знать степень каждой вершины, чтобы вычислить хроматическое число графа. В этом можно убедиться, рассматривая графы, приведенные на рис.8.3(а) и рис.8.3(б). Эти графы имеют по $n=12$ вершин, $m=16$ ребер и одинаковые распределения степеней вершин d_i . Однако хроматические числа данных графов равны 4 и 2 соответственно. При известных величинах n (число вершин), m (число ребер) и d_1, \dots, d_n (степени вершин графа) можно получить верхнюю и нижнюю оценки для хроматического числа графа. Задача нахождения хроматического числа произвольного графа является сложной проблемой, для решения которой не существует точного алгоритма решения. Для решения задачи о раскраске графа в настоящее время известны приближенные алгоритмы, позволяющие находить приближенное значение хроматического числа произвольного графа и соответствующую этому значению раскраску вершин.

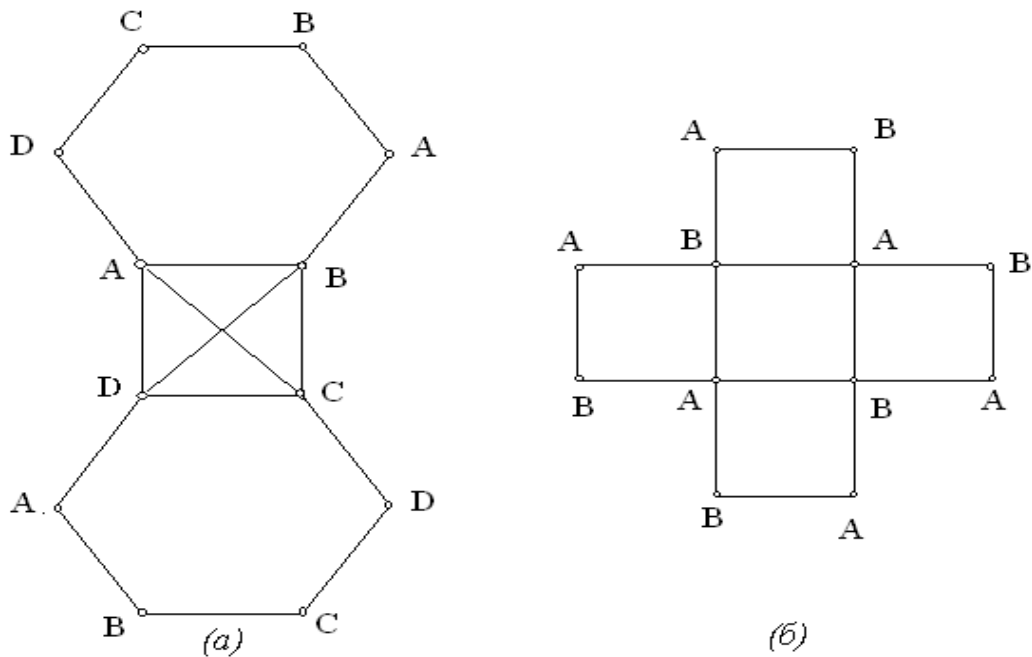


Рис.8.3. Графы с одинаковыми n , m и распределениями степеней вершин, но с различными хроматическими числами:
 (а) $\gamma(G)=4$. (б) $\gamma(G)=2$.

8.2. Нижние оценки для $\gamma(G)$

Поскольку число $\alpha(G)$ равно мощности наибольшего множества попарно несмежных вершин графа G , то оно совпадает также с мощностью наибольшего множества вершин в G , которые могут быть окрашены в один цвет, и, следовательно,

$$\gamma(G) \geq \left\lceil \frac{n}{\alpha(G)} \right\rceil, \quad (8.1)$$

где n — число вершин графа G , а $\lceil x \rceil$ обозначает наибольшее целое число, не превосходящее числа x . Еще одна нижняя оценка для $\gamma(G)$ предложена Геллером:

$$\gamma(G) \geq \frac{n^2}{n^2 - 2m}. \quad (8.2)$$

8.3. Верхние оценки для $\gamma(G)$

Нижние оценки хроматического числа, безусловно, более интересны, чем верхние, поскольку (если они достаточно близки к истинному значению) они могут быть использованы в процедуре вычисления $\gamma(G)$, включающей дерево поиска. В то же время верхние оценки хроматического числа подобного применения не находят. Тем не менее, в литературе приводятся формулы для вычис-

ления верхних оценок хроматического числа; так Бруксом предложена следующая легко вычисляемая оценка:

$$\chi(G) \leq 1 + \max_{x_i \in X} [d(x_i) + 1] \quad (8.3)$$

8.4. Приближенные алгоритмы раскрашивания

Существует много эвристических процедур раскрашивания графов, позволяющих находить хорошие приближения для хроматического числа графа в тех случаях, когда размеры графа слишком велики и получение оптимальной раскраски точными методами, упоминавшимися ранее, затруднительно. В настоящем разделе дается краткое описание одной из таких процедур и ряда ее разновидностей. Данная процедура относится к последовательным методам, основанным на упорядочивании множества вершин.

В этом простейшем из методов вершины вначале располагаются в порядке невозрастания их степеней. Первая вершина окрашивается в цвет 1; затем список вершин просматривается сверху вниз (по невозрастанию степеней) и в цвет 1 окрашивается всякая вершина, которая не смежна с другой вершиной, уже окрашенной, в этот цвет. Затем возвращаемся к первой в списке неокрашенной вершине, окрашиваем ее в цвет 2 и снова просматриваем список вершин сверху вниз, окрашивая в цвет 2 любую неокрашенную вершину, которая не смежна другой, уже окрашенной в цвет 2 вершине. Аналогично действуем с цветами 3, 4 и т.д., пока не будут окрашены все вершины. Число использованных цветов будет тогда приближенным значением хроматического числа графа.

Простая модификация описанной выше эвристической процедуры состоит в переупорядочивании неокрашенных вершин после окраски каждой очередной вершины: оставшиеся неокрашенные вершины записываются в порядке невозрастания их «относительных» степеней, т.е. степеней в таком графе, который получается из данного после удаления окрашенных вершин (вместе с ребрами, инцидентными удаленным вершинам).

В этой процедуре предполагалось, что если две вершины имеют одинаковые степени, то их взаимное положение в списке случайно. В таких ситуациях уточнение в размещении вершин можно осуществлять с помощью двухшаговых степеней $d_i^{(2)}$ вершин x_i , имеющих одинаковые степени (одинаковые 1-шаговые степени), где $d_i^{(2)}$ определяется как число маршрутов длины 2, исходящих из

x_i . Эти вершины могут быть размещены тогда в соответствии с величинами степеней $d_i^{(2)}$. Если все-таки найдутся вершины, у которых совпадают и степени d_i , и степени $d_i^{(2)}$, то можно вычислить трехшаговые степени $d_i^{(3)}$ (определяемые аналогичным образом) и разместить вершины с учетом степеней $d_i^{(3)}$ и т.д.

Можно действовать иначе: размещать вершины сразу в соответствии с их степенями $d_i^{(2)}$ или степенями $d_i^{(3)}$ и применять тот же самый последовательный метод раскраски. Таким образом, описанный выше метод раскрашивания очерчивает целый класс последовательных методов, каждый из которых связан с определенным способом упорядочивания вершин, либо статическим, т.е. фиксированным сразу для всей процедуры, либо динамическим, т.е. изменяющимся в процессе раскраски. Способ упорядочивания может базироваться на многих возможных критериях, зависящих от степеней вершин или от каких-либо других родственных характеристик.

Результаты вычислений и сравнение последовательных методов раскрашивания для графов, выбранных случайным образом, приведены в работах Матулы, Вильямса, Марбле и Исааксона. Границы применимости этих эвристических методов демонстрируются у Митчема, показавшего, что можно построить графы, для которых любой из эвристических методов дает произвольно плохие оценки хроматического числа.

9. ХРОМАТИЧЕСКИЕ ГРАФЫ

Теорема 9.1. Для того, что бы граф G был 2-хроматическим необходимо и достаточно, чтобы он был двудольным.

Доказательство 1. \Rightarrow Пусть G - 2-хроматический, тогда множество v вершин графа G разбито на два подмножества V_1 - вершины с цветом 1, V_2 - вершины с цветом 2. Отсюда следует, что G - двудольный граф.

2. \Leftarrow G -двудольный, тогда V_1 закрашиваем в цвет 1, V_2 - в цвет 2. Поскольку любое ребро $e = (u, v): u \in V_1, v \in V_2$, то G -2-хроматический.

Из теорем 3.1, 9.1 вытекает

Теорема 9.2. Для того чтобы граф G был 2-хроматическим, необходимо и достаточно, чтобы он не содержал циклов нечетной длины.

Подмножество V' вершин графа $G = (V, E)$ называется кликой, если любые две входящие в него вершины смежны.

Число вершин в максимальной клике графа G называется его плотностью (или кликовым числом) и обозначается через $\varphi(G)$.

Теорема 9.3. Граф G ρ -хроматический тогда и только тогда, когда он содержит ρ -вершинную клику K_ρ , которая является критическим подграфом G или стягивается к K_ρ .

Теорема 9.4. Любой критический подграф графа G такой, что $x(H) = x(G)$, является кликой или стягивается к K_ρ .

Теорема 9.5. Каждый критический подграф H_x графа G такой, что $x(H) = x(G)$, который не является кликой, получается из критического подграфа H_{x-1} добавлением к нему вершины v и ребер, соединяющих вершину v с каждой вершиной H_{x-1} .

Теорема 9.6. Граф G является ρ -хроматическим тогда и только тогда, когда он содержит критический подграф $H: x(H) = x(G)$ или $H = K_\rho$.

Теорема 9.7. Каждый граф G содержит критический подграф $H: x(H) = x(G)$.

Теорема 9.8. Каждый критический граф $H: x(H) = \chi(G)$ стягивается к K_ρ .

Гипотеза 1. Каждый n -хроматический связный граф G стягивается к K_n .

Гипотеза 2. Следующие два утверждения эквивалентны:

- 1) каждый n -хроматический граф стягивается к K_n .
- 2) гипотеза 4-х красок верна.

Критическим графом I рода будем называть критический граф H , который содержится в графе G , и выполняется равенство $x(H) = x(G)$.

Граф G будем называть критическим графом II рода, если граф G является критическим и стягивается к такому критическому подграфу $H: x(H) = x(G)$.

Известно, что каждый граф G содержит критический подграф $H: x(H) = x(G)$. Отсюда и из определений следует, что справедлива

Лемма 9.1. Каждый граф G содержит подграф, который является либо критическим графом I рода, либо критическим графом II рода.

Известна теорема Зыкова:

Теорема 9.9. Для любого целого $\gamma \geq 2$ существует простой (обыкновенный) граф G плотности $\varphi(G) = 2$ с произвольно большим хроматическим числом γ .

Доказательство. Для доказательства теоремы 9.9 индуктивно построим последовательность $S = (G_1, G_2, \dots, G_i)$ графов G_i без треугольников (рис.9.1.) таких, что $x(G_i) = i$. Положим $G_2 = K_2$.

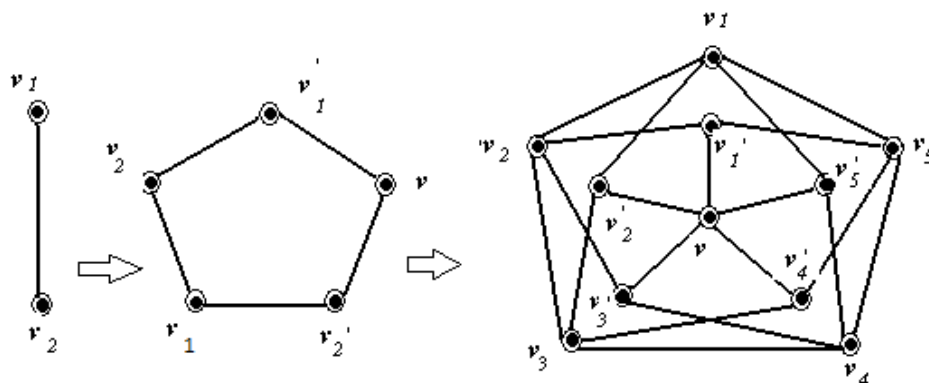


Рис.9.1. Последовательность графов без треугольников.

Если граф G_i уже построен, $i \geq 2$ и $V_i = \{v_1, v_2, \dots, v_n\}$. Добавив к G_i «дубликаты» вершин $V' = \{v'_1, v'_2, \dots, v'_n\}$, граф G_{i+1} определим по следующему правилу: $V_{i+1} = V_i \cup V' \cup v$, $V_i \cap V' = \emptyset$, $v \notin V_i \cup V'$, каждую вершину v'_j соединим ребрами с теми вершинами из V_i , с которыми смежна v_j в графе G_i ; вершину v соединим ребрами с каждой вершиной из V' . Полученный таким образом граф имеет $2n+1$ вершин.

Покажем, что G_{i+1} - искомый граф. Так как вершина v не смежна ни с одной вершиной из множества $V_i = \{v_1, v_2, \dots, v_n\}$, а вершины из $V' = \{v'_1, v'_2, \dots, v'_n\}$ попарно не смежны, то никакой треугольник не может содержать v . По той же причине треугольник не может содержать более одной вершины из множества V' , если треугольник образовывали вершины v_i, v_j, v_k , то эти вершины в графе G_i составляли бы треугольник. Поскольку в графе G_i треугольников нет, то в графе G_{i+1} также их нет. Кроме того, любую правильную i - раскраску графа G_i легко продолжить до правильной $(i+1)$ -раскраски графа G_{i+1} , положив $f(v'_j) = f(v_j)$ для $j=1, 2, \dots, n$ и приписав вершине v новый цвет. Таким образом, доказано, что граф G_{i+1} не содержит треугольников и $\chi(G_{i+1}) = i+1$. Теорема 9,9 доказана.

Из метода доказательства теоремы 9.9 следует, что полученная таким образом последовательность $S = \{G_i\}, i=1, 2, \dots, \gamma$ обладает следующими свойствами.

1⁰. Каждый граф G_i содержится в каждом из графов G_{i+1}, G_{i+2}, \dots , т.е. $G_2 \subset G_3 \subset G_4 \subset \dots \subset G_i \subset G_{i+1} \subset \dots \subset G_\gamma$.

2⁰. Графы $G_2, G_3, G_4, \dots, G_\gamma$ - критические I рода каждый.

3⁰. Граф $G_i, i \geq 5$ - содержит подграф, который является критическим II рода.

4⁰. Каждый граф $G'_i = G_i - v$ является критическим I рода.

5⁰. Каждый критический граф $G'_i = G_i - v$ является критическим подграфом Π рода, т.е. получается из графа G'_i стягиванием к центрам всех ребер, содержащих центральные вершины со всеми вершинами их окружения.

6⁰. Каждый критический граф Π рода G'_i стягивается к K_i .

Алгоритм нахождения хроматического числа

1. Ввести k - число подмножеств $V_i, i=1,2,\dots,k$.
2. Выбрать множество V_s .
3. Выбрать множество $V_m, m \neq s$.
4. Выбрать вершину $v_i \notin V_s$.
5. На вход подать список N_i ; Вычислить $p = |N_i|$.
6. Исключить v_i из V_s , т.е. $V_s := V_s \setminus \{v_i\}$.
7. Выбрать вершину $v_j \in N_i$.
8. Исключить вершины v_j из N_i , т.е. $N_i = N_i \setminus \{v_j\}$.
9. Если $N_i \neq \emptyset$ и $v_j \in N_i$, $v_j \in V_m$, то $r := r + 1; v_j = v_m$ {выбрать следующие вершины списка N_i }; перейти к п. 6.
10. Если $N_i \neq \emptyset$ и $v_j \in N_i$, $v_j \notin V_m$, то $r := 0; v_j = v_m$ {перейти к следующим вершинам списка N_i } перейти к п. 6.
11. Если $r \geq 1$, то $t := t + 1$; выбрать следующее множество V_e , т.е. $V_m := V_e$, перейти к п. 3.
12. Если $t < k - 1$, то { v_i - не является ps -вершиной} вершина v_i исключается из множества V_m и включается в такое множество V_p , что $V_p \subset \{v_i\}$ - независимое множество; перейти к следующим вершинам множества V_s $v_i := v_k$; перейти к п. 2. Если $V_m \neq \emptyset$, то $k := k - 1$ {перейти к следующему множеству $V_m := V_p$; $v_i := v_e$: перейти к п. 3}.
13. Если $t = k - 1$, то { v_i - ps -вершина, $s := s + 1$ } пометить ps -вершины.
14. Если $s < k$, то {в списке N_i ps - вершины v_i не все вершины являются ps -вершинами}

15. В списке $N(v_i)$ выбрать не помеченные вершины v_s . Найти множество $V_m, m \neq s$, для которого $V_m \cup v_s$ – независимое множество. Исключить вершины $v_s \in V_s$ из V_s и включить в множество V_m : {перейти к следующим вершинам множества V_s }; $v_i = v_e$; перейти к п 2.
16. Если $V_s \neq \emptyset$, то $k = k - 1, s := s + 1$.
17. Если $s < k$, то перейти к п 1.
18. Если $\rho = k$, то { ps - вершина смежна хотя бы одной ps - вершине каждого множества $v_i, i = 1, 2, \dots, k$ }.
19. Вывести множества V_i, k .

10. ПРИМЕНЕНИЯ РАСКРАСКИ ГРАФА

Задача раскраски в том «чистом» виде, в каком она рассматривалась выше в настоящей лекции, редко встречается на практике. Однако ее обобщения и разновидности находят широкое применение в большом числе различных прикладных задач. Целью данного раздела является ознакомление читателя с несколькими наиболее часто встречающимися обобщениями. Список приложений, естественно, этими примерами не ограничивается.

10.1. Простая задача размещения (загрузки)

Рассмотрим задачу размещения (загрузки) n каких-то предметов по ящикам. Пусть каждый предмет соответствует определенной вершине графа G . Всякий раз, когда два предмета x_i и x_j не могут быть размещены в одном ящике (например, когда предмет x_i может загрязнить предмет x_j), в граф G вводится ребро (x_i, x_j) .

Если ящики имеют неограниченную вместимость такую, что в каждый из них можно поместить сколько угодно предметов, то задача нахождения наименьшего числа ящиков для размещения предметов эквивалентна задаче нахождения хроматического числа графа G . При этом каждому ящику соответствует определенный «цвет», а предметы, окрашенные в один цвет, укладываются в один и тот же ящик.

10.2. Составление графиков осмотра (проверки)

В задачах теории расписаний осмотра представляются в виде временных интервалов. Каждому осмотру можно сопоставить вершину некоторого графа, причем две любые вершины графа будут соединены ребром лишь тогда, когда соответствующие им осмотры нельзя осуществлять одновременно. Требуется составить такой график осмотра, который связан с наименьшими временными затратами (с учетом приведенных выше ограничений на «совместимость» осмотров). Эта задача эквивалентна задач о раскраске вершин графа с использованием наименьшего числа цветов. Хроматическое число графа как раз и соответствует осмотру, требующему наименьших временных затрат.

10.3. Задача распределения ресурсов

Пусть для выполнения каких-то n работ надо распределить m имеющихся в наличии ресурсов. Считаем, что каждая из работ вы-

полняется за некоторый (одинаковый для всех работ) промежуток времени и что для выполнения i -й работы требуется подмножество ресурсов S_i . Построим граф G : каждой работе соответствует определенная вершина графа, а ребро (x_i, x_j) существует в графе тогда и только тогда, когда для выполнения i -й и j -й работ требуется хотя бы один общий ресурс, т. е. когда $S_i \cap S_j \neq \emptyset$. Это означает, что i -я и j -я работы не могут выполняться одновременно. Раскраска графа G определяет тогда некоторое распределение ресурсов (по выполняемым работам), причем такое, что работы, соответствующие вершинам одного цвета, выполняются одновременно. Наилучшее использование ресурсов (т.е. выполнение всех n работ за наименьшее время) достигается при оптимальной раскраске вершин графа G .

Задачи и упражнения

1. Найдите хроматическое число всех графов, изображенных на рисунках этого пособия.
2. Для любого n приведите пример графа с хроматическим числом n .
3. Чему равно хроматическое число дерева?
4. Чему равно хроматическое число простой цепи?
5. Чему равно хроматическое число простого цикла?
6. Составить программу раскраски графа (используя алгоритм последовательной раскраски).
7. Верно ли утверждение «граф G является r -хроматическим тогда и только тогда, когда он содержит r -вершинную клику K_r »?

КОМПЛЕКТ ЗАДАНИЙ ДЛЯ ТЕСТИРОВАНИЯ по дисциплине «Дискретная математика»

1. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 5\}$, $B=\{2,4,5,6\}$, $C=\{1,3,5,6\}$.

Найти $A \cup B$ (Указать правильные варианты ответов).

- a. $\{1,2,2,3,4,4,5,6\}$
- b. $\{1,2,3,4,5,6\}$ (+3 балла)
- c. $\{x \mid x < 7, x \in U\}$ (+4 балла)
- d. $\{1,3\}$
- e. $\{3,4,2,5,1,6\}$ (+3 балла)

2. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 4\}$, $B=\{2,4,5,7\}$, $C=\{1,2,5,6\}$.

Найти $C \cup A$ (Указать правильные варианты ответов).

- a. $\{1,1,2,2,3,5,6\}$
- b. $\{1,2,3,5,6\}$ (+5 баллов)
- c. $\{x \mid x < 7\}$
- d. $\{3,2,6,1,5\}$ (+5 баллов)
- e. $\{1,2\}$

3. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x > 4\}$, $B=\{3,5,7\}$, $C=\{1,2,4,6\}$.

Найти $C \cup B$ (Указать правильные варианты ответов).

- a. U (+4 балла)
- b. $\{3,5,7\}$
- c. \emptyset
- d. $\{3,5,7,1,2,4,6\}$ (+3 балла)
- e. $\{1,2,3,4,5,6,7\}$ (+3 балла)

4. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 5\}$, $B=\{2,4,5,6\}$, $C=\{1,3,5,6\}$.

Найти $C \cap B$ (Указать правильные варианты ответов).

- a. $\{1,2,3,4,5,5,6,6\}$
- b. $\{6,5\}$ (+5 баллов)
- c. $\{1,2,3,4,5,6\}$
- d. $\{x \mid x < 7\}$
- e. $\{5,6\}$ (+5 баллов)

5. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 4\}$, $B=\{2,4,5,7\}$, $C=\{1,2,5,6\}$. Найти $A \cap B$ (Указать правильные варианты ответов).

- a. $\{1,2,3,4,5,7\}$
- b. $\{1,2,2,3,4,5,7\}$
- c. $\{2\}$ (+5 баллов)
- d. $\{5,6\}$
- e. $\{x \mid x=2\}$ (+5 баллов)

6. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x > 4\}$, $B=\{3,5,7\}$, $C=\{1,2,4,6\}$.

Найти $B \cap A$ (Указать правильные варианты ответов).

- a. $\{7,5\}$ (+5 баллов)
- b. $\{3,5,6,7\}$
- c. $\{5,7,5,7\}$
- d. $\{5,7\}$ (+5 баллов)
- e. $\{x \mid 2 < x < 8\}$

7. Тип - дистрибутивный вопрос

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 5\}$, $B=\{2,4,5,6\}$, $C=\{1,3,5,6\}$.

Найти декартово (прямое) произведение $D \times C$, где $D = A - B$ (Указать правильные варианты ответов).

- a. $\{1,3,5,6\}$
- b. $\{(1,1), (3,1), (1,3), (3,3), (1,5), (3,5), (1,6), (3,6)\}$ (+6 баллов)
- c. $\{(1,1), (1,3), (3,3), (1,5), (3,5), (1,6), (3,6)\}$
- d. $\{(1,3), (1,5), (3,5), (1,6), (3,6)\}$
- e. $\{(3,3), (1,5), (3,5), (1,6), (3,6), (1,1), (3,1), (1,3)\}$ (+6 баллов)
- f. $\{1,1,3,3,5,6\}$

8. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x < 4\}$, $B=\{2,4,5,7\}$, $C=\{1,2,5,6\}$.

Найти декартово (прямое) произведение $D \times A$, где $D = C - B$ (Указать правильные варианты ответов).

- a. $\{1,2,3,6\}$
- b. $\{(1,1), (6,1), (1,2), (6,2), (1,3), (6,3)\}$ (+4 балла)
- c. $\{(1,1), (1,6), (1,2), (2,6), (1,3), (3,6)\}$
- d. $\{1\}$
- e. $\{(1,1), (1,2), (1,3), (6,1), (6,2), (6,3)\}$ (+4 балла)

f. $\{(6,3), (1,1), (1,3), (6,1), (6,2), (1,2)\}$ (+4 балла)

9. Тип - дистрибутивный вопрос.

Дано универсальное множество $U=\{1,2,3,4,5,6,7\}$ и в нем подмножества $A=\{x \mid x > 4\}$, $B=\{3,5,7\}$, $C=\{1,2,4,6\}$. Найти декартово (прямое) произведение $B \times D$, где $D=C-A$ (Указать правильные варианты ответов).

Варианты ответов:

a. $\{1,2,3,4,5,7\}$

b. $\{(3,1), (5,1), (7,1), (3,2), (5,2), (7,2), (3,4), (5,4), (7,4)\}$
(+6 баллов)

c. $U - \{4\}$

d. $\{(1,3), (2,3), (3,4), (1,5), (2,5), (4,5), (1,7), (2,7), (4,7)\}$

e. $\{(3,1), (3,2), (3,4), (5,1), (5,2), (5,4), (7,1), (7,2), (7,4)\}$
(+6 баллов)

f. \emptyset

10. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A - (B \cup C) = (A - B) \cup (A - C)$$

a. да

b. нет (+5 баллов)

11. Тип - альтернативный вопрос

Справедлив ли дистрибутивный закон?

$$A \otimes (B \cup C) = (A \otimes B) \cup (A \otimes C)$$

a. да

b. нет (+5 баллов)

12. Тип - альтернативный вопрос

Справедлив ли дистрибутивный закон?

$$A(B - C) = AB - AC$$

a. да (+5 баллов)

b. нет

#Ответ# да# (+5 баллов)

13. Тип - альтернативный вопрос

Справедлив ли дистрибутивный закон?

$$A \cup BC = (A \cup B)(A \cup C)$$

a. да(+5 баллов)

b. нет

14. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A - (B \cap C) = (A - B) \cap (A - C)$$

а. да

б. нет (+5 баллов)

15. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A \otimes B \cap C = (A \otimes B) \cap (A \otimes C)$$

а. да

б. нет (+5 баллов)

16. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A - (B \otimes C) = (A - B) \otimes (A - C)$$

а. да

б. нет (+5 баллов)

17. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A(B \otimes C) = AB \otimes AC$$

а. да (+5 баллов)

б. нет

18. Тип - альтернативный вопрос.

Справедлив ли дистрибутивный закон?

$$A \cup (B - C) = (A \cup B) - (A \cup C)$$

а. да

б. нет (+5 баллов)

19. Тип - простой вопрос.

Сколькими способами можно выбрать 3 различных карандаша из имеющихся 5 карандашей разных цветов? (Ввести ответ в виде числа)

#Ответ# 10# (+10 баллов)

20. Тип - простой вопрос.

Сколькими способами можно разделить 5 различных карандашей между двумя школьниками так, чтобы у каждого был хотя бы один карандаш? (Ввести ответ в виде числа)

#Ответ# 30# (+10 баллов)

21. Тип - простой вопрос.

Сколькими способами можно разделить 8 шахматистов на две команды по 4 человека? (Ввести ответ в виде числа)

#Ответ# 35# (+10 баллов)

22. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Найти радиус $r(G)$ графа.

#Ответ# 3# (+10 баллов)

23. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 0 & 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{pmatrix}$$

Найти диаметр $d(G)$ графа.

#Ответ# 4# (+10 баллов)

24. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Найти радиус $r(G)$ графа.

#Ответ# 2# (+10 баллов)

25. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$$

Найти диаметр $d(G)$ графа.

#Ответ# 2# (+10 баллов)

26. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Найти радиус $r(G)$ графа.

#Ответ# 2# (+10 баллов)

27. Тип - простой вопрос.

Граф G задан следующей матрицей смежности:

$$\begin{pmatrix} 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$$

Найти диаметр $d(G)$ графа.

#Ответ# 3# (+10 баллов)

28. Тип - простой вопрос.

Сколько существует неизоморфных деревьев с 6 вершинами?

#Ответ# 6# (+10 баллов)

29. Тип - простой вопрос.

Сколько существует неизоморфных связных графов с 5 вершинами и 4 ребрами?

#Ответ# 3# (+10 баллов)

30. Тип - простой вопрос.

Сколько существует неизоморфных связных графов с 5 вершинами и 5 ребрами?

#Ответ# 5# (+10 баллов)

31. Тип - дистрибутивный вопрос.

Выберите условия, каждое из которых является необходимым для того, чтобы связный граф с n вершинами был планарным (m – число ребер):

- a. $m \leq 3n - 6$ (+3 балла)
- b. $m < 3n - 6$
- c. $m = 8$ при $n = 6$
- d. $m < 19$ при $n = 8$ (+4 балла)
- e. $m \leq 3n$ (+3 балла)

32. Тип - дистрибутивный вопрос.

Выберите условия, каждое из которых является достаточным для того, чтобы граф с n вершинами был планарным (m – число ребер):

- a. $m \leq 3n - 6$
- b. граф не содержит подграфа, гомеоморфного графу $K_{3,3}$, и подграфа, гомеоморфного графу K_5 (+3 балла)
- c. $m = n - 1$, и граф связный (+4 балла)
- d. граф не содержит подграфа, изоморфного графу $K_{3,3}$
- e. $m = 5$ при $n = 7$ (+3 балла)

33. Тип - дистрибутивный вопрос.

Выберите условия, каждое из которых является достаточным для того, чтобы граф с n вершинами не был планарным (m – число ребер):

- a. граф содержит подграф, изоморфный графу K_5 (+2 балла)
- b. $m = 10$ при $n = 20$
- c. граф содержит подграф, гомеоморфный графу K_6 (+3 балла)
- d. $m > 3n$ (+2 балла)
- e. $m = 10$ при $n = 5$ (+3 балла)

34. Тип - дистрибутивный вопрос.

Пусть граф G с n вершинами является деревом. Тогда: (Выберите для G верные утверждения)

- a. число ребер $m = n - 1$ (+2 балла)
- b. граф связный (+3 балла)
- c. граф не содержит циклов (+2 балла)

- d. граф планарный (+2 балла)
- e. граф не эйлеров (+2 балла)
- f. есть вершина степени 1 (+3 балла)
- g. есть вершина степени больше 1

35. Тип - дистрибутивный вопрос.

Пусть граф G с n вершинами является несвязным. Тогда: (Выберите для G верные утверждения.)

- a. число компонент связности всегда равно 2
- b. число компонент связности может быть равно 2 (+3 балла)
- c. степень каждой вершины не превосходит $n - 2$ (+4 балла)
- d. число компонент связности больше 1 (+3 балла)
- e. граф не может быть двудольным
- f. граф планарный
- g. граф не может быть деревом (+4 балла)

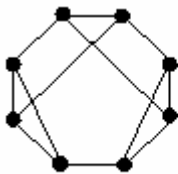
36. Тип - дистрибутивный вопрос.

Пусть граф G с n вершинами является двудольным. Тогда: (Выберите для G верные утверждения.)

- a. в нем нет циклов четной длины
- b. в нем могут быть циклы четной длины (+7 баллов)
- c. в нем все циклы имеют четную длину (+7 баллов)
- d. граф связный
- e. степень каждой вершины не превосходит $n - 2$
- f. граф содержит цикл, если каждая доля содержит не менее двух вершин
- g. граф планарный

37. Тип - альтернативный вопрос.

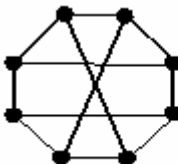
Является ли планарным следующий граф:



- a. да (+5 баллов)
- b. нет

38. Тип - альтернативный вопрос.

Является ли планарным следующий граф:



- a. да
- b. нет (+5 баллов)

39. Тип - альтернативный вопрос.

Является ли планарным следующий граф:



- a. да (+5 баллов)
- b. нет

40. Тип - альтернативный вопрос.

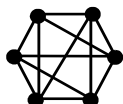
Является ли планарным следующий граф:



- a. да (+5 баллов)
- b. нет

41. Тип - альтернативный вопрос.

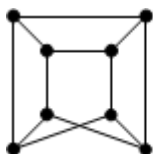
Является ли планарным следующий граф:



- a. да (+5 баллов)
- b. нет

42. Тип - альтернативный вопрос.

Является ли планарным следующий граф:



- a. да
- b. нет (+5 баллов)

43. Тип - простой вопрос.

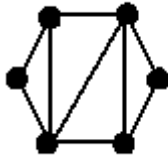
Сколько граней у плоского графа:



#Ответ# 4# (+5 баллов)

44. Тип - простой вопрос.

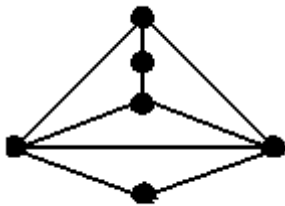
Сколько граней у плоского графа:



#Ответ# 5# (+5 баллов)

45. Тип - простой вопрос.

Сколько граней у плоского графа:



#Ответ# 5# (+5 баллов)

46. Тип - простой вопрос.

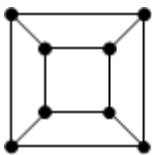
Сколько граней у плоского графа:



#Ответ# 8# (+5 баллов)

47. Тип - простой вопрос.

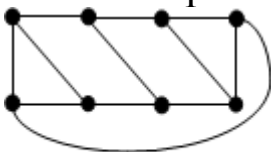
Сколько граней у плоского графа:



#Ответ# 6# (+5 баллов)

48. Тип - простой вопрос.

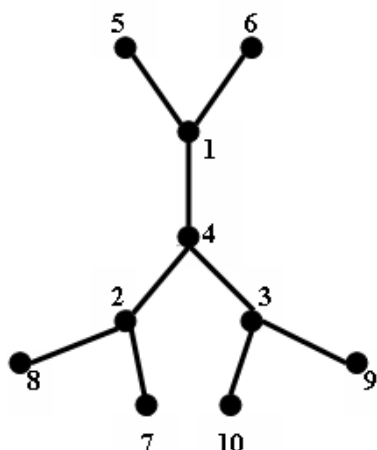
Сколько граней у плоского графа:



#Ответ# 6# (+5 баллов)

49. Тип - альтернативный вопрос.

По дереву найти соответствующий ему код Прюфера $P(t)$
(Указать его вариант).



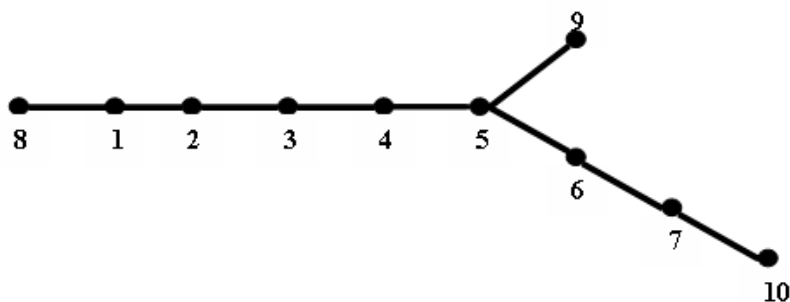
a. $P(t) = (2\ 2\ 1\ 1\ 4\ 4\ 3\ 3)$

b. $P(t) = (1\ 2\ 1\ 2\ 3\ 4\ 3\ 4)$

c. $P(t) = (1\ 1\ 4\ 2\ 2\ 4\ 3\ 3)$ (+10 баллов)

50. Тип - альтернативный вопрос.

По дереву найти соответствующий ему код Прюфера $P(t)$
(Указать его вариант).



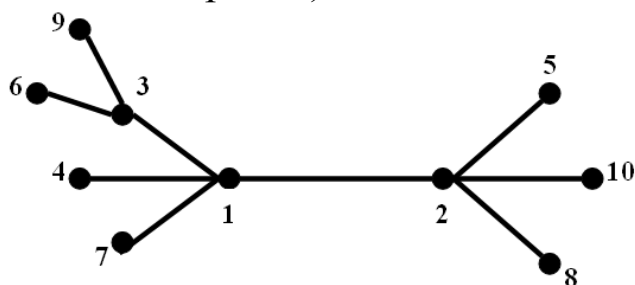
a. $P(t) = (1\ 2\ 3\ 4\ 5\ 6\ 6\ 7)$

b. $P(t) = (1\ 2\ 3\ 4\ 5\ 5\ 6\ 7)$ (+10 баллов)

c. $P(t) = (1\ 2\ 3\ 4\ 5\ 6\ 7\ 7)$

51. Тип - альтернативный вопрос.

По дереву найти соответствующий ему код Прюфера $P(t)$
(Указать его вариант).



a. $P(t) = (1\ 1\ 1\ 2\ 2\ 2\ 3\ 3)$

b. $P(t) = (3\ 3\ 1\ 1\ 1\ 2\ 2\ 2)$

с. $P(t) = (1\ 2\ 3\ 1\ 2\ 3\ 1\ 2)$ (+10 баллов)

52. Тип - дистрибутивный вопрос.

Для функции f , заданной вектором $\alpha_f = (0111)$, определить, является ли она:

- а. линейной
- б. монотонной (+5 баллов)
- с. самодвойственной
- д. функцией из класса T_0 (+5 баллов)
- е. функцией из класса T_1 (+5 баллов)

53. Тип - дистрибутивный вопрос.

Для функции f , заданной вектором $\alpha_f = (0110)$, определить, является ли она:

- а. линейной (+8 баллов)
- б. монотонной
- с. самодвойственной
- д. функцией из класса T_0 (+7 баллов)
- е. функцией из класса T_1

54. Тип - дистрибутивный вопрос.

Для функции f , заданной вектором $\alpha_f = (1011)$, определить, является ли она:

- а. нелинейной (+8 баллов)
- б. монотонной
- с. самодвойственной
- д. функцией из класса T_0
- е. функцией из класса T_1 (+7 баллов)

55. Тип - дистрибутивный вопрос.

Для функции $f = x \oplus y \oplus z$ определить, является ли она:

- а. линейной (+5 баллов)
- б. монотонной
- с. самодвойственной (+5 баллов)
- д. функцией из класса T_0 (+5 баллов)
- е. функцией из класса T_1 (+5 баллов)

56. Тип - дистрибутивный вопрос.

Для функции $f = xy \oplus z \oplus 1$ определить, является ли она:

- а. линейной
- б. немонотонной (+10 баллов)
- с. самодвойственной

- d. функцией из класса T_0
- e. функцией из класса T_1 (+10 баллов)

57. Тип - дистрибутивный вопрос.

Для функции $f = xy \oplus xz$ определить, является ли она:

- a. линейной
- b. монотонной
- c. несамодвойственной (+10 баллов)
- d. функцией из класса T_0 (+10 баллов)
- e. функцией из класса T_1

58. Тип - альтернативный вопрос.

Полна ли система функций $\{f, g, h\}$ (принадлежность функций классам T_0, T_1, L, M, S отображена в таблице).

Функции	T_0	T_1	L	M	S
f	+	-	+	+	-
g	-	+	+	+	-
h	+	+	-	+	+

- a. да
- b. нет (+5 баллов)

59. Тип - альтернативный вопрос.

Полна ли система функций $\{F, G, H\}$ (принадлежность функций классам T_0, T_1, L, M, S отображена в таблице).

Функции	T_0	T_1	L	M	S
F	-	+	-	-	-
G	-	+	+	+	-
H	-	-	-	-	+

- a. да (+5 баллов)
- b. нет

60. Тип - альтернативный вопрос.

Полна ли система функций $\{f, g, h\}$ (принадлежность функций классам T_0, T_1, L, M, S отображена в таблице).

Функции	T_0	T_1	L	M	S
f	-	-	+	-	+
g	+	+	+	+	+
h	+	+	-	-	+

- a. да
- b. нет (+5 баллов)

61. Тип - альтернативный вопрос.

Верно ли, что:

$$T_0 S \subseteq T_1$$

а. да (+10 баллов)

б. нет

62. Тип - альтернативный вопрос.

Верно ли, что:

$$T_0 T_1 L \subseteq S$$

а. да (+10 баллов)

б. нет

63. Тип - альтернативный вопрос.

Верно ли, что:

$$MS \subseteq T_0$$

а. да (+10 баллов)

б. нет

Критерии оценки:

91-100 % верных ответов – «отлично»;

81-90% верных ответов – «хорошо»;

50-80% верных ответов – «удовлетворительно»;

Меньше 50% верных ответов – «неудовлетворительно»;

КОМПЛЕКТ ВОПРОСОВ К ЭКЗАМЕНУ

1. Высказывания и операции над ними.
2. Основные эквивалентности (законы) алгебры высказываний.
3. Вычисление и упрощение логических выражений ,
4. Методы доказательств.
5. Метод математической индукции.
6. Множества. Операции над множествами.
7. Способы задания множеств.
8. Разбиения и покрытия.
9. Алгебра подмножеств. Булеан.
10. Свойства операций над множествами.
11. . Объединение конфигураций.
12. Классическая формула метода включений- исключений.
13. Отношения. Прямое произведение множеств.
14. Композиция отношений.
15. Степень и ядро отношения.
16. Свойства отношений.
17. Представление отношений в ЭВМ.
18. Функции (отображения).
19. Инъекция, сюръекция и биекция.
20. Индуцированная функция.
21. Принцип Дирихле.
22. Отношение эквивалентности.
23. Классы эквивалентности.
24. Фактор-множества.
25. Отношения порядка. Минимальные элементы
26. Замыкание отношений.
27. Транзитивное и рефлексивное транзитивное замыкание.
28. Числовые последовательности.
29. Рекуррентное соотношение.
30. Суммируемые последовательности. Способы нахождения некоторых сумм.
31. Суммы и рекуррентности.
32. Бином Ньютона. Биномиальные коэффициенты.
33. Основные тождества с биномиальными коэффициентами.
34. Полиномиальная формула. Полиномиальные коэффициенты.
35. Правило суммы.
36. Правило произведения.

37. Асимптотические методы решений рекуррентных соотношений.
38. Вычисление суммы бесконечной геометрической прогрессии.
39. Метод суммирования Эйлера.
40. Метод производящих функций. Числа Фибоначчи.
41. Метод включения и исключения. Теорема.
42. Основные определения и понятия теории графов.
43. Теорема о сумме степеней вершин графа.
44. Теорема о числе вершин нечетной степени в графе.
45. Подграфы.
46. Дополнение графа.
47. Изоморфные графы.
48. Маршруты, цепи, циклы.
49. Вершинно - порожденные графы.
50. Реберно-порожденные графы.
51. Операции над графами.
52. Разбиение n -множества.
53. Связные графы. Теорема.
54. Компоненты связности графа.
55. Двудольные графы. Критерий двудольности графа.
56. Метрические характеристики графа.
57. Алгоритм поиска в ширину.
58. Применение алгоритма поиска в ширину.
59. Нахождение эксцентриситета вершины.
60. Нахождение диаметра и радиуса графа.
61. Эйлеровы графы. Теорема.
62. Алгоритм нахождения Эйлера цикла в графе.
63. Гамильтоновы графы. Задача коммивояжера.
64. Плоские графы.
65. Планарные графы.
66. Грани плоского графа. Формула Эйлера.
67. Гомеоморфные графы.
68. Теорема Понтрягина - Куратовского.
69. Раскрашиваемость вершин двудольного графа.

ЗАКЛЮЧЕНИЕ

Теория графов находит широкое применение в различных областях науки и техники:

Графы и информация

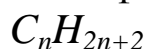
Двоичные деревья играют весьма важную роль в теории информации. Предположим, что определенное число сообщений требуется закодировать в виде конечных последовательностей различной длины, состоящих из нулей и единиц. Если вероятности кодовых слов заданы, то наилучшим считается код, в котором средняя длина слов минимальна по сравнению с прочими распределениями вероятности. Задачу о построении такого оптимального кода позволяет решить алгоритм Хаффмана.

Двоичные кодовые деревья допускают интерпретацию в рамках теории поиска. Каждой вершине при этом сопоставляется вопрос, ответить на который можно либо "да", либо "нет". Утвердительному и отрицательному ответу соответствуют два ребра, выходящие из вершины. "Опрос" завершается, когда удастся установить то, что требовалось.

Таким образом, если кому-то понадобится взять интервью у различных людей, и ответ на очередной вопрос будет зависеть от заранее неизвестного ответа на предыдущий вопрос, то план такого интервью можно представить в виде двоичного дерева.

Графы и химия

Еще А. Кэли рассмотрел задачу о возможных структурах насыщенных (или предельных) углеводородов, молекулы которых задаются формулой:



Молекула каждого предельного углеводорода представляет собой дерево. Если удалить все атомы водорода, то оставшиеся атомы углеводорода также будут образовывать дерево, каждая вершина которого имеет степень не выше 4. Следовательно, число возможных структур предельных углеводородов, т. е. число гомологов данного вещества, равно числу деревьев с вершинами степени не больше четырех.

Таким образом, подсчет числа гомологов предельных углеводородов также приводит к задаче о перечислении деревьев определенного типа. Эту задачу и ее обобщения рассмотрел Д. Пойа.

Графы и биология

Деревья играют большую роль в биологической теории ветвящихся процессов. Для простоты мы рассмотрим только одну разновидность ветвящихся процессов – размножение бактерий. Предположим, что через определенный промежуток времени каждая бактерия либо делится на две новые, либо погибает. Тогда для потомства одной бактерии мы получим двоичное дерево.

Нас будет интересовать лишь один вопрос: в скольких случаях n -е поколение одной бактерии насчитывает ровно k потомков? Рекуррентное соотношение, обозначающее число необходимых случаев, известно в биологии под названием процесса Гальтона-Ватсона. Его можно рассматривать как частный случай многих общих формул.

Графы и физика

Еще недавно одной из наиболее сложных и утомительных задач для радиолюбителей было конструирование печатных схем.

Печатной схемой называют пластинку из какого-либо диэлектрика (изолирующего материала), на которой в виде металлических полосок вытравлены дорожки. Пересекаться дорожки могут только в определенных точках, куда устанавливаются необходимые элементы (диоды, триоды, резисторы и другие), их пересечение в других местах вызовет замыкание электрической цепи. В ходе решения этой задачи необходимо вычертить плоский граф, с вершинами в указанных точках.

Итак, из всего вышесказанного неопровержимо следует практическая ценность теории графов.

СПИСОК ЛИТЕРАТУРЫ

Основная

1. Емеличев В.А., Перепелица В.А., Козырев В.А. Обзор некоторых проблем дискретной многокритериальной оптимизации.//Труды сем. по дискретной математике и ее прилож.- М.: МГУ,1989.- С.13-17.
2. Емеличев В.А., Мельников О.И., Сарванов В.И., Тышкевич Р.И. Лекции по теории графов. М.: Наука, 1990, 384 с.
3. Берж К. Теория графов и ее применения. - М.: Иностран. лит., 1962.
4. Асанов М.О., Баранский В.А., Расин В.В. Дискретная математика: графы, матроиды, алгоритмы. - Ижевск: НИЦ "РХД",2001.
5. Ахо А., Хопкрофт Дж. Построение и анализ вычислительных алгоритмов.- М.:Мир,1979.
6. Гермейер Ю.Б. Введение в теорию исследования операций. М.: Наука, 1971.
7. Евстигнеев В.А. Применение теории графов в программировании. М.: Наука, 1985.
8. Ковалев М.М. Дискретная оптимизация. - Минск: Изд-во БГУ, 1977.
9. Майника Э. Алгоритмы оптимизации на сетях и графах. М.:Мир, 1981.
10. Марков А.А., Нагорный Н.М. Теория алгоритмов. М.: Наука, 1982.
11. Михалевич В.С., Волкович В.Л. Вычислительные методы исследования и проектирования сложных систем.- М.: Наука, 1982.
12. Моисеев Н.Н. Математические задачи системного анализа. М.: Наука, 1979.
13. Оре О. Теория графов. М.: Наука, 1980.
14. Пападимитриу Х., Стайглиц К. Комбинаторная оптимизация. Алгоритмы и сложность.- М.:Мир. 1985.
15. Саати Т. Целочисленные методы оптимизации и связанные с ними экстремальные проблемы. М.:Мир. 1973.
16. Сергиенко И.В. Математические модели и методы решения задач дискретной оптимизации.- Киев: Наукова думка, 1985.

17. Свами М., Тхуласираман К. Графы, сети и алгоритмы. М.: Мир, 1984.
18. Татт У. Теория графов. М.: Мир, 1988.
19. Уилсон Р. Введение в теорию графов. М.: Мир, 1977.
20. Успенский В.А., Семенов А.Л. Теория алгоритмов: основные открытия и приложения. М.: Наука, 1987.
21. Харари Ф. Теория графов. М.: Мир, 1973.
22. Зыков А.А. Основы теории графов. М.: Наука, 1987
23. Гэри Д., Джонсон М. Вычислительные машины и труднорешаемые задачи. М.: Мир, 1985.- 412с.
24. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. – М.: Энергоатомиздат, 1988.
25. Кук Д., Бейз Г. Компьютерная математика. – М.: Наука, 1990.
26. Нефедов В.Н., Осипова В.А. Курс дискретной математики. – М.: Издательство МАИ, 1992.
27. Оре О. Теория графов. – М.: Наука, 1980.
28. Исмагилов Р.С., Калинин А.В. Материалы к практическим занятиям по курсу: Дискретная математика по теме: Алгоритмы на графах. - М.: МГТУ, 1995.
29. Смольяков Э.Р. Введение в теорию графов. М.: МГТУ, 1992
30. Нечепуренко М.И. Алгоритмы и программы решения задач на графах и сетях. - Новосибирск: Наука, 1990.
31. Романовский И.В. Алгоритмы решения экстремальных задач. - М.: Наука, 1977.
32. Писсанецки С. Технология разреженных матриц. - М.: Мир, 1988.
33. Севастьянов Б.А. Вероятностные модели. - М.: Наука, 1992.
34. Бочаров П.П., Печинкин А.В. Теория вероятностей. - М.: Изд-во РУДН, 1994.
35. Шунгаров Х. Д. К решению задачи разбиения дерева на цепи длины два. В сб. науч.тр. Всероссийского симпозиума «Информационные технологии в учебном процессе», Карачаевск, 2009с.
36. Хаггарти Р. Дискретная математика для программистов. М.: Техносфера, 2003.-320 с.
37. Меньшиков М.В., Ревякин А.М., Копылова А.Н., Макарова Ю.Н., Стечкин Б.С. Комбинаторный анализ. Задачи и упражнения. Учебное пособие/ Под ред. К.А. Рыбникова. М.: Наука, 1982.-368с.

38. Яблонский, С.В. Введение в дискретную математику: учеб. пособие для вузов/ С.В. Яблонский. - 3-е изд., стер. - М.: Высш. шк., 2001. - 384с.
39. Сачков В.Н. Введение в комбинаторные методы дискретной математики. М.:Наука.1982.-284с.
40. Кнут Д. Искусство программирования для ЭВМ. Т.1. Основные алгоритмы. М: Мир,1976. Т. II. Получисленные алгоритмы. М: Мир,1977. Т. III. Сортировка и поиск. М: Мир,1978.
41. Кузнецов О.П. Дискретная математика для инженера/ О.П. Кузнецов. - Изд. 3-е, перераб. и доп. - СПб. Лань, 2004. - 394 с.
42. Новиков Ф.А. Дискретная математика для программистов. /СПб: Питер. 2000. -304с.
43. Спирина, М.С. Дискретная математика: учебник/ М.С. Спирина, П.А. Спирин. - М.: АСADEMIA, 2004. - 367 с.
44. Пугач Л.И. Высшая математика. Задачи по дискретной математике, математической логике и теории алгоритмов : метод. указания к практическим занятиям для студентов 1 курса специальностей 220400 "Программное обеспечение", "22300 " Системы автоматизированного проектирования" и 075300 "Организация и технология защиты информации"/ Л.И. Пугач ; БГТУ. - Брянск: Изд-во БГТУ, 2005. - 16 с.
45. Асеев Г.Г. Дискретная математика : учеб. пособие/ Г.Г. Асеев, О.М. Абрамов, Д.Э. Ситников. - Ростов н/Д; Харьков: Феникс: Торсинг, 2003. - 141 с.
46. Акимов О.Е. Дискретная математика: логика, группы, графы. – М.: Лаборатория Базовых Знаний, 2001. – 376 с.
47. Лавров И. А., Максимова Л. Л. Задачи по теории множеств, математической логике и теории алгоритмов.— М.: Физ. Мат.лит., 2004. -256 с.
48. Новиков Ф.А. Дискретная математика для программистов. - С.-Пб. Питер, 2001. – 304 с.
49. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. - М.: Энергоатомиздат, 1988 – 480с

Дополнительная

1. Аляев Ю.А. Тюрин С.Ф. Дискретная математика и математическая логика. — М.: Финансы и статистика, 2006. — 368 с.
2. Андерсон, Джеймс А. Дискретная математика и комбинаторика. - Пер. с англ. — М.: Издатель - Издательский дом "Вильямс", 2004. — 960 с.
3. Виленкин Н.Я Комбинаторика. - М., Наука, 1969. -328 с.
4. Белоусов, А.И. Дискретная математика: учебник/ А.И. Белоусов, С.Б. Ткачев; под ред. В.С. Зарубина, А.П. Крищенко. - М.: МГТУ им Н.Э. Баумана, 2001. - . вып. 19. - 743с.
5. Горбатов, В.А. Основы дискретной математики: учеб. пособие для вузов/ В.А. Горбатов. - М.: Высш. шк., 1986. - 310 с.
6. Горбатов, В.А Дискретная математика: учебник для студентов втузов/

**Дискретная математика
(теория, решение задач, программирование)**

Учебное пособие

План университета 2017, поз. 1

Редактор	Н.В. Ефрюкова
Корректор	Х.Д. Шунгаров
Компьютерная вёрстка и набор	С.А. Бостанова

Подписано в печать 20.01.2017

Формат 60x84/16

Бумага офисная

Объем: 6,5 усл. печ.л.

Тираж 100 экз.

**Издательство Карачаево-Черкесского государственного
университета им. У.Д. Алиева: 369202,
г. Карачаевск, ул. Ленина, 29
Лицензия ЛР № 040310 от 21.10.1997.**

**Отпечатано в типографии Карачаево-Черкесского
государственного университета им. У.Д. Алиева: 369202.
г. Карачаевск, ул. Ленина, 46**

