

УЧЕБНОЕ ПОСОБИЕ  
ПО  
ТЕМАТИКЕ  
«ГРАФЫ И СЕТИ»

Е. В. Харитонова

# ГРАФЫ И СЕТИ

Ульяновск 2006

ФЕДЕРАЛЬНОЕ АГЕНТСТВО ПО ОБРАЗОВАНИЮ  
Государственное образовательное учреждение высшего профессионального образования  
Ульяновский государственный технический университет

**Е. В. Харитонова**

# **ГРАФЫ И СЕТИ**

Учебное пособие

для студентов экономических специальностей

Ульяновск 2006

УДК 512.5+333 (075)

ББК 22.17я7

X20

Рецензенты:

Кафедра общенаучных дисциплин ИАТУ УлГТУ (зав. кафедрой канд. хим. наук М. В. Бебякова, доцент А. Г. Карасева); канд. физ.-мат. наук кафедры экономико-математических методов и информационных технологий Института экономики и бизнеса УлГУ А. Е. Эткин

Утверждено редакционно-издательским советом университета в качестве учебного пособия.

**Харитонов, Е. В.**

X 20 Графы и сети : учебное пособие / Е. В. Харитонов. – Ульяновск : УлГТУ, 2006. – 92 с.

ISBN 5-89146-878-6

В учебном пособии изложены основы теории графов и сетей. Материал каждого раздела проиллюстрирован примерами и сопровождается подборкой задач для практических занятий, имеются контрольные задания. Содержание пособия соответствует государственному образовательному стандарту высшего образования для экономических специальностей.

Пособие рекомендовано для специальностей 080109 «Бухгалтерский учет, анализ и аудит», 080507 «Менеджмент организации» и может использоваться студентами и преподавателями при подготовке к занятиям.

**УДК 512.5+333 (075)**

**ББК 22.17я7**

Учебное издание  
ХАРИТОНОВА Елена Викторовна

**ГРАФЫ И СЕТИ**

Учебное пособие

Редактор *А. В. Власова*

Подписано в печать 03.06.2006. Формат 60×84/16.

Бумага офсетная. Печать трафаретная. Усл. печ. л. 5,29.

Тираж 100 экз. Заказ .

Ульяновский государственный технический университет

432027, г. Ульяновск, ул. Сев. Венец, 32

Типография УлГТУ, 432027, г. Ульяновск, ул. Сев. Венец, 32

ISBN 5-89146-878-6

© Харитонов Е. В.

© Оформление. УлГТУ, 2006

## ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ .....	4
1. ГРАФЫ .....	5
1.1. Основные понятия и определения .....	5
1.2 Ориентированные графы.....	10
1.3 Пути и циклы Эйлера .....	14
1.4 Планарные графы.....	15
1.5 Пути и циклы Гамильтона .....	17
1.6 Алгоритм поиска кратчайшего пути.....	19
УПРАЖНЕНИЯ .....	23
2. СЕТИ.....	32
2. 1. Сети и потоки .....	32
2.2 Сети Петри.....	46
УПРАЖНЕНИЯ .....	53
3. ОСНОВНЫЕ ТИПЫ ЗАДАЧ И СПОСОБЫ ИХ РЕШЕНИЯ .....	57
3.1 Расчет временных параметров сетевого графика .....	59
3.2 Построение сетевого графика и распределение ресурсов .....	62
3.3 Учет стоимостных факторов при реализации сетевого графика .....	64
3.4 Обоснование привлекательности проекта по выпуску продукции .....	67
3.5 Минимизация сети .....	72
УПРАЖНЕНИЯ .....	76
ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ .....	81
ОТВЕТЫ К УПРАЖНЕНИЯМ .....	85
ЗАКЛЮЧЕНИЕ .....	91
БИБЛИОГРАФИЧЕСКИЙ СПИСОК.....	92

## ВВЕДЕНИЕ

Первой работой теории графов как математической дисциплины считают статью Эйлера (1736 г.), в которой рассматривалась задача о Кёнингсбергских мостах. Эйлер показал, что нельзя обойти сеть городских мостов и вернуться в исходную точку, пройдя по каждому мосту ровно один раз. Следующий импульс теория графов получила спустя почти 100 лет с развитием исследований по электрическим сетям, кристаллографии, органической химии и другим наукам.

С графами, сами того не замечая, мы сталкиваемся постоянно. Например, графом является схема линий метрополитена. Точками на ней представлены станции, а линиями – пути движения поездов. Исследуя свою родословную и возводя ее к далекому предку, мы строим так называемое генеалогическое древо. И это древо – граф.

Графы служат удобным средством описания связей между объектами. Например, рассматривая граф, изображающий сеть дорог между населенными пунктами, можно определить маршрут проезда от пункта *A* до пункта *B*. Если таких маршрутов окажется несколько, хотелось бы выбрать в определенном смысле оптимальный, например самый короткий или самый безопасный. Для решения задачи выбора требуется проводить определенные вычисления над графами. При решении подобных задач удобно использовать алгебраическую технику, да и само понятие графа необходимо формализовать.

Методы теории графов широко применяются в дискретной математике. Без них невозможно обойтись при анализе и синтезе различных дискретных преобразователей.

В настоящее время теория графов охватывает большой материал и активно развивается.

До появления сетевых методов планирование работ, проектов осуществлялось в небольшом объеме. Наиболее известным средством такого планирования был ленточный график Ганта, недостаток которого состоит в том, что он не позволяет установить зависимости между различными операциями.

Современное сетевое планирование начинается с разбиения программы работ на операции. Определяются оценки продолжительности операций, и строится сетевая модель (график). Построение сетевой модели позволяет проанализировать все операции и внести улучшения в структуру модели до начала ее реализации. Строится календарный график, определяющий начало и окончание каждой операции, а также взаимосвязи с другими операциями графика. Календарный график выявляет критические операции, которым надо уделять особое внимание, чтобы закончить все работы в директивный срок. Что касается некритических операций, то календарный план позволяет определить резервы времени, которые можно выгодно использовать при задержке выполнения работ или эффективном применении как трудовых, так и финансовых ресурсов.

# 1. ГРАФЫ

## 1.1. Основные понятия и определения

• Граф есть конечное множество  $V$ , называемое множеством вершин, и множество  $E$  двухэлементных подмножеств множества  $V$ . Множество  $E$  называется *множеством ребер*. Элемент множества  $E$  называется *ребром*. Граф обозначается  $G(V, E)$ . Элементы  $a$  и  $b$  элементы множества  $V$  называются *соединенными* или *связанными* ребром  $\{a, b\}$ , если  $\{a, b\} \in E$ .

Обычно конечный граф изображают в виде диаграммы, на которой вершины обозначаются точками, а ребра, соединяющие две вершины, – линиями между этими точками.

• Если  $\{a, b\}$  – ребро, тогда вершины  $a$  и  $b$  называются *концами* ребра  $\{a, b\}$ . Ребро  $\{a, b\}$  называют также *инцидентным* к вершинам  $a$  и  $b$ .

• Две вершины называются *смежными*, если они являются концами ребра, или, что то же самое, если они инцидентны к одному ребру.

• Два ребра называются *смежными*, если они инцидентны к общей вершине.

**Пример.** Граф с множеством вершин  $V = \{a, b, c\}$  и множеством ребер  $E = \{\{a, b\}, \{b, c\}\}$  может быть изображен, как показано на рис.1.1 или рис.1.2.



Рис. 1.1

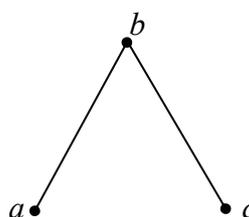


Рис. 1.2

**Пример.** Граф, у которого  $V = \{a, b, c, d, e\}$  и  $E = \{\{a, b\}, \{a, e\}, \{b, e\}, \{b, d\}, \{b, c\}, \{c, d\}\}$  может быть изображен диаграммой, показанной на рис. 1.3.

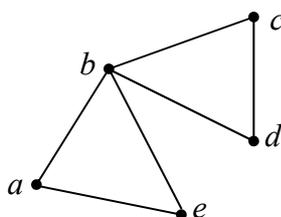


Рис. 1.3

- Графы, определенные выше, называют *простыми графами*.
- Ребро, которое соединяет вершину саму с собой, называется *петлей*.

Если в графе допускается наличие петель, то он называется *графом с петлями*.

Если в графе допускается наличие более одного ребра между двумя вершинами, то он называется *мультиграфом*.

Если каждая вершина графа отмечена, то граф называется *размеченным*.

Если же допускается как наличие петель, так и существование более одного ребра между двумя вершинами, то такой объект называют *псевдографом*.

- *Степенью* вершины  $v$ , обозначается  $\deg(v)$ , называется количество ребер, инцидентных этой вершине. Вершина степени 0 называется *изолированной*.

**Пример.** В графе, показанном на рис. 1.4, вершины  $a$  и  $c$  – смежные, а  $e_1, e_2$  и  $e_3$  – смежные ребра. Однако, вершины  $a$  и  $f$  смежными не являются, а  $e_2$  и  $e_5$  не являются смежными ребрами. Вершины  $b, c, d$  имеют степень 2, в то время как вершины  $a$  и  $f$  имеют степень 3.

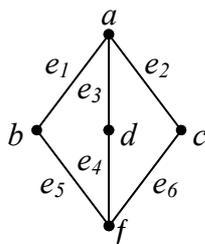


Рис. 1.4

**ТЕОРЕМА 1.1.** Сумма степеней вершин графа всегда четная.

**Доказательство.** Поскольку каждое ребро графа имеет два конца, степень каждого конца увеличивается на 1 за счет одного ребра. Таким образом, в сумму степеней всех вершин каждое ребро вносит 2 единицы, поэтому сумма должна в два раза превышать число ребер. Следовательно, сумма является четным числом.

**ТЕОРЕМА 1.2.** В любом графе количество вершин нечетной степени чётно.

**Доказательство.** Доказательство проведем методом от противного: предположив, что теорема не верна, найдем противоречие, из которого будет следовать, что теорема справедлива. Если теорема не верна, то имеется нечетное количество вершин, степени которых нечетны. Но сумма степеней вершин с четными степенями четна. Сумма степеней всех вершин есть сумма степеней вершин с нечетными степенями плюс сумма степеней вершин с четными степенями. Поскольку сумма нечетного числа и четного числа есть число нечетное,

сумма степеней всех вершин нечетная. Но это противоречит теореме 1.1., поэтому мы пришли к противоречию. Следовательно, теорема справедлива.

• Граф  $G(V', E')$  называется *подграфом* графа  $G(V, E)$ , обозначается  $G'(V', E') \prec G(V, E)$ , если  $V' \subseteq V, E' \subseteq E$ .

Таким образом, каждая вершина в  $G'$  является вершиной в  $G$ , и каждое ребро в  $G'$  является ребром в  $G$ .

**Пример.** Графы, изображенные на рис. 1.5, 1.6, 1.7, являются подграфами графа на рис.1.8.

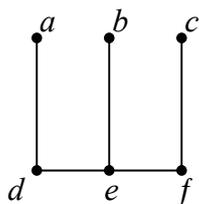


Рис. 1.5

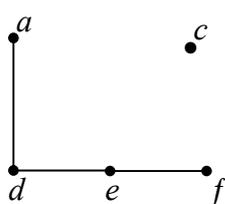


Рис. 1.6

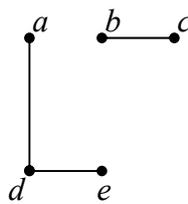


Рис. 1.7

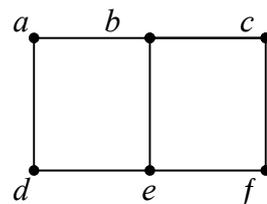


Рис. 1.8

Путь (маршрут) в графе – это совокупность ребер, которые объединены вместе с вершинами так, что вдоль них можно двигаться по графу. Для удобства договоримся использовать символы  $v_0, v_1, v_2, v_3, \dots, v_k$ ,  $u$  и  $v$  для обозначения вершин.

• Пусть  $G = G(V, E)$  – граф с вершинами  $v_0, v_1, \dots, v_k \in V$  и ребрами  $e_1, e_2, \dots, e_k \in E$ .

Путем длины  $k$  из  $v_0$  в  $v_k$  (или между  $v_0$  и  $v_k$ ) называется последовательность  $v_0 e_1 v_1 e_2 v_2 e_3 v_3 \dots v_k e_k v_k$  такая, что  $e_i = \{v_{i-1} v_i\}$ . Путь длины  $k$  имеет  $k$  ребер.

В общем случае путь будет обозначаться через  $v_0, v_1, v_2, \dots, v_k$ .

Каждые два последовательных ребра пути имеют общую вершину, поэтому являются смежными.

• *Простым путем* из  $v_0$  в  $v_k$  называется путь, в котором нет повторяющихся вершин.

**Пример.** В графе, изображенном на рис. 1.9. Из  $v_0$  в  $v_7$  ведут пути  $v_0 v_1 v_2 v_5 v_7$ ,

$$v_0 v_1 v_2 v_5 v_4 v_1 v_2 v_5 v_7,$$

$$v_0 v_1 v_4 v_5 v_4 v_5 v_7,$$

$$v_0 v_3 v_4 v_6 v_7$$

длины 4, 8, 6 и 4 соответственно. Пути  $v_0 v_1 v_2 v_5 v_7$  и  $v_0 v_3 v_4 v_6 v_7$  являются простыми.

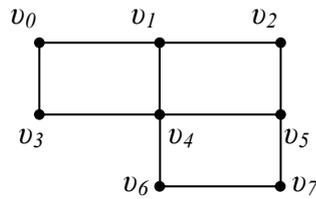


Рис. 1.9

• Граф  $G$  называется *связным*, если имеется путь между любыми двумя его различными вершинами.

**Пример.** Граф на рис. 1.10 не связный. Например, нет пути из  $v_0$  в  $v_3$  и между  $v_2$  и  $v_4$ .

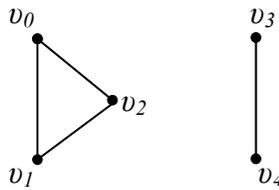


Рис. 1.10

Возвращаясь к рассмотрению графа на рис. 1.9, следует отметить, что путь  $v_0 v_1 v_2 v_5 v_4 v_1 v_2 v_5 v_7$  можно сократить до  $v_0 v_1 v_2 v_5 v_7$ . Поскольку вершина  $v_1$  повторялась, необходимо удалить часть пути между двумя появлениями вершины  $v_1$  и после первого появления вершины  $v_1$  переходить сразу к  $v_2$ .

Таким образом, если путь включает какую-либо вершину  $v_i$  более чем один раз, его можно сократить, удалив  $v_i$  и вершины, лежащие на пути между двумя появлениями вершины  $v_i$ . Проведенные рассуждения дают возможность сформулировать следующую теорему.

**ТЕОРЕМА 1.3.** Пусть  $G = G(V, E)$  – граф. Если существует путь из вершины  $v_i$  в вершину  $v_j$ , тогда существует простой путь из вершины  $v_i$  в вершину  $v_j$ .

Комбинируя определение связного графа и теорему 1.3, приходим непосредственно к следствию:

**Следствие.** Граф  $G$  является связным тогда и только тогда, когда между любыми двумя его вершинами существует простой путь.

• Пусть  $G = G(V, E)$  – граф. Подграф  $G'$  графа  $G$  называется *компонентой* графа  $G$ , если выполнены следующие два условия:

1.  $G'$  – непустой связный граф.
2.  $G''$  – связный подграф графа  $G$  и  $G' \preceq G''$ , тогда  $G = G''$ . Отсюда  $G''$  – максимальный связный подграф графа  $G$ .

**Пример.** Графы, изображенные на рис. 1.11 и 1.12 являются компонентами графа на рис. 1.10.

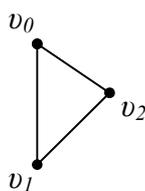


Рис. 1.11.



Рис. 1.12.

- Пусть  $G = G(V, E)$  – граф. *Циклом* называется путь нулевой длины, соединяющий вершину саму с собой и не содержащий повторяющихся ребер.

- *Простым циклом* называется цикл, соединяющий вершину  $v$  саму с собой и не содержащий повторяющихся вершин, кроме  $v$ .

- Цикл называется  $n$  – *циклом*, если он содержит  $n$  ребер и  $n$  различных вершин.

**Пример.** Если рассмотреть граф на рис. 1.9, то в нем пути  $v_0 v_1 v_4 v_3 v_0$ ,

$v_0 v_1 v_4 v_5 v_7 v_6 v_4 v_3 v_0$ ,

$v_1 v_2 v_5 v_7 v_6 v_4 v_1$ ,

$v_0 v_1 v_2 v_5 v_7 v_6 v_4 v_3 v_0$

являются циклами. При этом все циклы, кроме  $v_0 v_1 v_4 v_5 v_7 v_6 v_4 v_3 v_0$ , простые.

- Граф называется *полным*, если любые его две вершины соединены ребром.

Полный граф с  $n$  вершинами обозначается  $K_n$ .

**Пример.** На рис. 1.13 показаны, соответственно, графы  $K_2, K_3, K_4, K_5$

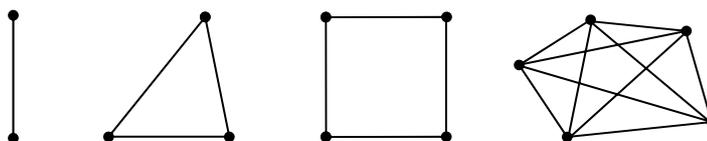


Рис. 1.13.

- Граф  $G = G(V, E)$  называется *двудольным*, если  $V$  можно представить как объединение непересекающихся множеств, скажем,  $V = A \cup B$ , так что каждое ребро имеет вид  $\{a, b\}$ , где  $a \in A$  и  $b \in B$ .

Таким образом, каждое ребро связывает вершину из  $A$  с вершиной из  $B$ , но никакие две вершины из  $A$  или две вершины из  $B$  не являются связанными.

- Двудольный граф называется *полным двудольным* графом  $K_{m,n}$ , если  $A$  содержит  $m$  вершин,  $B$  содержит  $n$  вершин и для каждого  $a \in A, b \in B$  имеем  $\{a, b\} \in E$ .

Таким образом, для каждого  $a \in A$  и  $b \in B$  имеется связывающее их ребро.

**Пример.** Графы  $K_{1,2}, K_{2,3}, K_{2,2}, K_{3,3}$  приведены по порядку на рис. 1.14.

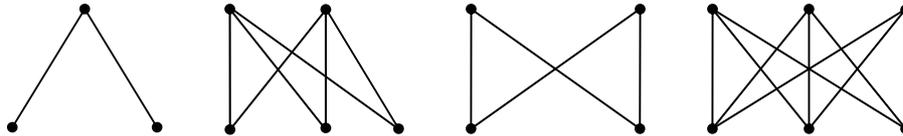


Рис. 1.14.

## 1.2 Ориентированные графы

Во многих случаях необходимы графы, у которых ребра, по существу, представляют собой улицу с односторонним движением. Это означает, что если рассматривается ребро, выходящее из вершины  $a$  в вершину  $b$ , то нельзя рассматривать выходящим из вершины  $b$  в вершину  $a$ .

- *Ориентированный граф* или *орграф*  $G$ , который обозначается через  $G(V, E)$ , состоит из множества  $V$  вершин и множества  $E$  упорядоченных пар элементов из  $V$ , называемого множеством *ориентированных ребер* или просто ребер.

- Элемент множества  $E$  называется *ориентированным ребром*.

- Если  $(a, b) \in E$ , то  $a$  называется *начальной* вершиной ребра  $(a, b)$ , а  $b$  называется *конечной* вершиной.

Понятие ориентированного графа допускает наличие петель, чего не было в случае простых графов.

Ребро  $(a, b)$  ориентированного графа обозначается на диаграмме стрелкой из  $a$  в  $b$ .

Отметим, что в простом графе ребро представляется двухэлементным подмножеством, чтобы подчеркнуть, что отношение симметрично, в то время как в ориентированном графе ребро представлено упорядоченной парой, чтобы акцентировать важность порядка и то, что  $(a, b)$  может быть ребром в орграфе, а  $(b, a)$  – нет.

**Пример.** Орграф, у которого  $V = \{a, b, c\}$  и  $E = \{(a, b), (b, c), (c, b), (c, a)\}$  изображен на рис. 1.15

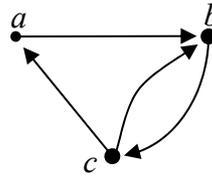


Рис. 1.15

**Пример.** Орграф, у которого  $V = \{a, b, c, d\}$  и  $E = \{(a, b), (b, c), (c, c), (b, d), (d, b), (c, d), (d, a)\}$  изображен на рис. 1.16.

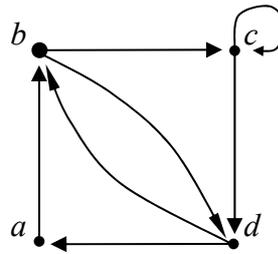


Рис. 1.16

- Если  $(a, b)$  – ребро ориентированного графа  $G(V, E)$ , так что  $a$  – начальная вершина, а  $b$  – конечная вершина ребра  $(a, b)$  тогда вершины  $a$  и  $b$  *инцидентны* ребру  $(a, b)$ . Вершина  $a$  называется *смежной* к вершине  $b$ . Вершина  $b$  называется также смежной от вершины  $a$ .

- *Степенью выхода* вершины  $v$  называется количество ребер, для которых  $v$  является начальной вершиной, обозначается  $\text{outdeg}(v)$ .

- *Степенью входа* вершины  $v$  называется количество ребер, для которых  $v$  является конечной вершиной, обозначается  $\text{indeg}(u)$ .

- Если  $\text{indeg}(v) = 0$ , то вершина  $v$  называется *источником*. Если  $\text{outdeg}(v) = 0$ , то вершина  $v$  называется *стоком*.

**Пример.** В ориентированном графе на рис. 1.17,  $\text{indeg}(v_0) = 0$ ,  $\text{indeg}(v_1) = 1$ ,  $\text{indeg}(v_2) = 2$ ,  $\text{indeg}(v_3) = 2$ ,  $\text{indeg}(v_4) = 3$ , так же  $\text{outdeg}(v_0) = 3$ ,  $\text{outdeg}(v_1) = 2$ ,  $\text{outdeg}(v_2) = 2$ ,  $\text{outdeg}(v_3) = 1$ ,  $\text{outdeg}(v_4) = 0$ . Вершина  $v_0$  – источник,  $v_4$  – сток.

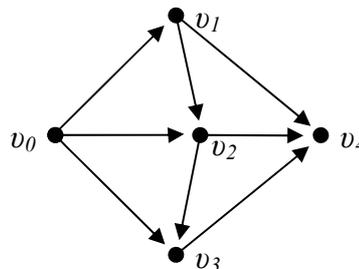


Рис. 1.17

- Ориентированный граф с более чем одним ребром из одной вершины в другую называется *мультиграфом* или *ориентированным мультиграфом*.

- Если каждое ребро помечено, то говорят, что это *размеченный ориентированный* или просто *размеченный граф*, с пониманием того, что это ориентированный граф.

- *Размеченный граф*  $G = G(V, L, E)$  представляет собой множество вершин  $V$ , множество меток  $L$  и множество  $E$ , которое является подмножеством  $V \times L \times V$ .

Таким образом, ребро  $e$  графа  $G$  имеет вид  $(a, l, b)$ , где  $l$  – метка, а  $a, b$  – вершины.

Графически ребро  $e = (a, l, b)$  размеченного графа обозначается, как на рис. 1.18, или, как на рис. 1.19, если ребро – петля.

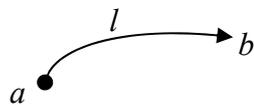


Рис. 1.18.



Рис. 1.19.

Графы, представленные на рис. 1.20 и 1.21, являются примерами типичных размеченных графов, называемых автоматами.

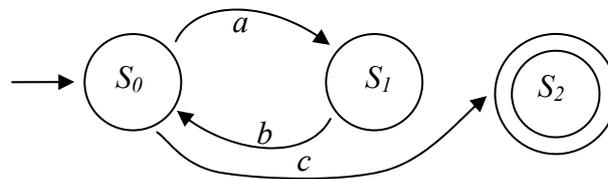


Рис. 1.20

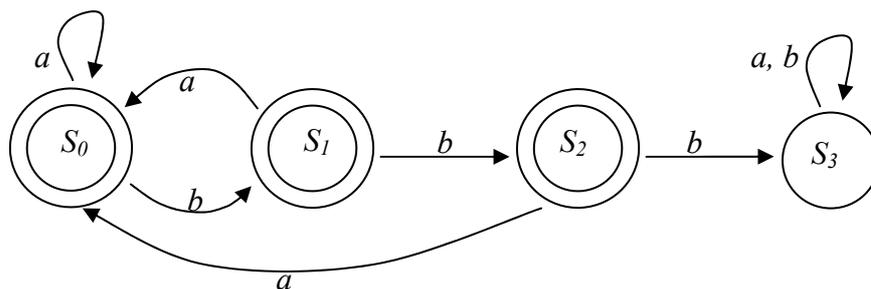


Рис. 1.21

- Ориентированный граф  $G'(V', E')$  называется *ориентированным подграфом* ориентированного графа  $G(V, E)$ , обозначается  $G'(V', E') \preceq G(V, E)$ , если  $V' \subseteq V, E' \subseteq E$ .

Таким образом, каждая вершина в  $G'$  является вершиной в  $G$  и каждое ориентированное ребро в  $G'$  является ориентированным ребром в  $G$ .

- *Ориентированный путь* из  $a$  в  $b$  задается последовательностью вершин  $v_0 v_1 v_2 \dots v_n$ , где  $a = v_0$ ,  $b = v_n$  и  $(v_{i-1}, v_i)$  для  $1 \leq i \leq n$  – ориентированное ребро.

- *Длиной* ориентированного пути называется количество ориентированных ребер, входящих путь.

**Пример.** Для графа  $G$ , изображенного на рис. 1.22, графы на рис. 1.23 и 1.24 являются подграфами. Ориентированные пути в  $G$  включают  $v_0 v_1 v_2 v_4$ ,  $v_1 v_2 v_4$  и  $v_0 v_3 v_4$ .

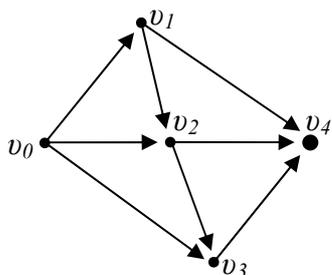


Рис. 1.22

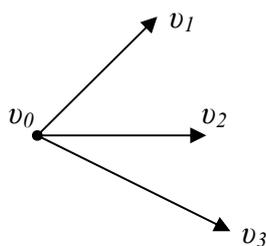


Рис. 1.23

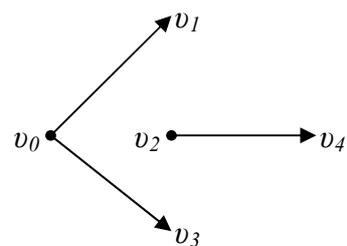


Рис.1.24

Теперь для заданного ориентированного графа  $G$  будет построен неориентированный граф  $G^s$  такой, что каждое ориентированное ребро  $G$  (исключая петли) станет неориентированным ребром графа  $G^s$ .

Пусть для каждого ориентированного графа  $G(V, E)$ ,  $E' = E - \{(v, v) : v \in V\}$ , так что  $G(V, E')$  – ориентированный подграф графа  $G(V, E)$ , в котором удалены петли. Пусть  $R$  – симметричное замыкание множества  $E'$ , так что если  $(a, b) \in E'$ , то  $(a, b), (b, a) \in R$ , а  $E^s$  – множество ребер, представляющих отношение  $R$ . В таком случае граф  $G^s(V, E^s)$  называется *соотнесенным графом* ориентированного графа  $G(V, E)$ .

- Ориентированный граф  $G(V, E)$  называется *связным*, если его соотнесенный граф является связным.

- Ориентированный граф называется *сильно связным*, если для любой пары вершин  $a, b \in V$  существует ориентированный путь из  $a$  в  $b$ .

**Пример.** Для ориентированного графа на рис. 1.22 соотнесенным будет граф на рис. 1.25.

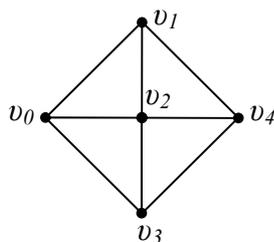


Рис. 1.25

Ориентированный граф является связным, поскольку его соотнесенный граф связный. Рассматриваемый граф, однако не является сильно связным, поскольку из  $v_1$  в  $v_3$  не существует ориентированного пути.

### 1.3 Пути и циклы Эйлера

• Пусть  $G = G(V, E)$  – граф. Цикл, который включает все ребра и вершины графа  $G$ , называется *эйлеровым циклом*.

Если это условие выполняется, говорят, что граф  $G$  имеет эйлеров цикл.

**ТЕОРЕМА 1.4.** Граф с более чем одной вершиной имеет эйлеров цикл тогда и только тогда, когда он связный и каждая его вершина имеет четную степень.

**Пример.** Граф на рис. 1.26 имеет эйлеров цикл, поскольку степень каждой его вершины четная, граф на рис. 1.27 не имеет эйлерова цикла, поскольку степени вершины  $v_2$  и  $v_4$  – нечетные.

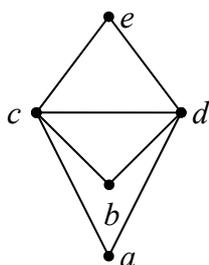


Рис. 1.26

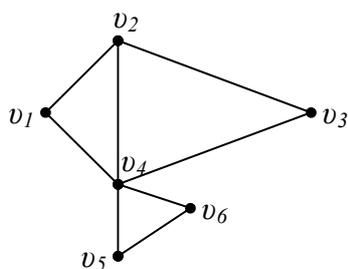


Рис. 1.27

• Пусть  $G = G(V, E)$  – граф. Путь, который включает каждое ребро графа  $G$  только один раз называется *эйлеровым путем*.

В этом случае говорят, что граф  $G$  имеет *эйлеров путь*.

• Если эйлеров путь не является эйлеровым циклом, то такой путь называют *собственным эйлеровым путем*.

**ТЕОРЕМА 1.5.** Граф (мультиграф или псевдограф) имеет собственный эйлеров путь тогда и только тогда, когда он связный и ровно две его вершины имеют нечетную степень.

**Пример.** Граф на рис. 1.28 имеет собственный эйлеров путь, т.к. ровно две его вершины имеют нечетную степень. Граф на рис. 1.29 не имеет собственного эйлерова пути, т.к. четыре его вершины имеют нечетную степень.

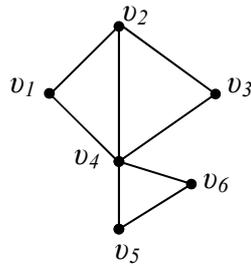


Рис. 1.28

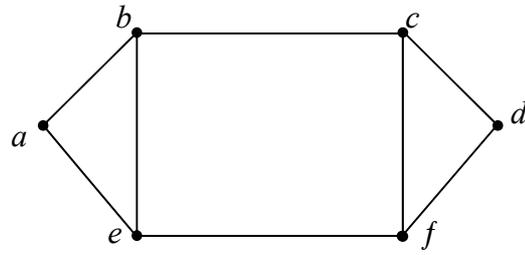


Рис. 1.29

- Пусть  $G = G(V, E)$  – ориентированный граф.  
*Ориентированным циклом* называется ориентированный путь ненулевой длины из вершины в ту же вершину без повторения ребер.
- Пусть  $G = G(V, E)$  – ориентированный граф.  
 Ориентированный цикл, которого включает все ребра и вершины графа  $G$ , называется *эйлеровым циклом*.

**ТЕОРЕМА 1.6.** Ориентированный граф имеет эйлеров цикл тогда и только тогда, когда он связный и степень входа каждой вершины равна ее степени выхода.

**Пример.** Ориентированный граф на рис. 1.30 имеет эйлеров цикл, так как степень входа каждой вершины равна степени выхода. Ориентированный граф на рис. 1.31 не имеет эйлерова цикла, так как степень входа вершины  $v_1$  не равна ее степени выхода.

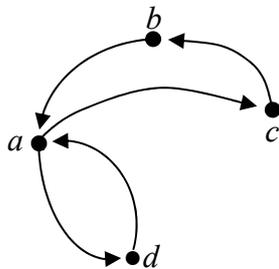


Рис. 1.30

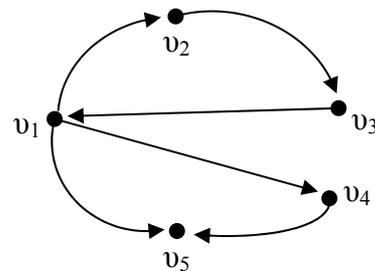


Рис. 1.31

## 1.4 Планарные графы

Интегральная микросхема состоит из слоев миниатюрных микросхем, впечатанных в пластину. В такой ситуации важно исключить пересечение проводов в местах, не предназначенных для соединений. Если изобразить места указанных соединений вершинами графа, то возникает задача построения графа с непересекающимися ребрами. Важно отметить, что интересует возможность построения графа с непересекающимися ребрами.

**Например,** граф, изображенный на рис. 1.32, изображается также другим способом (рис. 1.33). Граф, изображенный на рис. 1.34, может быть изображен, как показано на рис. 1.35.

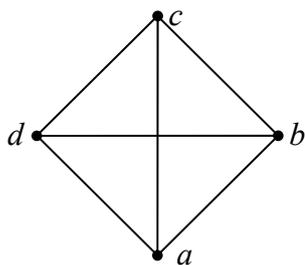


Рис. 1.32

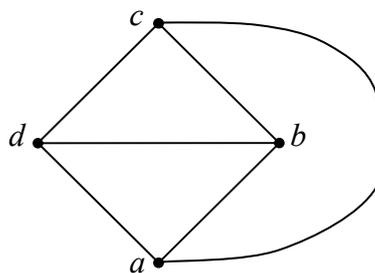


Рис. 1.33

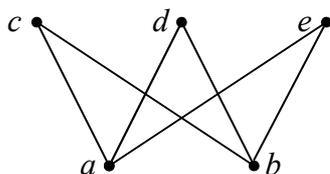


Рис. 1.34

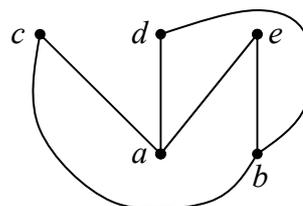


Рис. 1.35

- **Планарным графом** называется граф, который может быть изображен в плоскости, так что его ребра не пересекаются. Граф, который не является планарным, называется *непланарным*.

Рассмотрим граф как рисунок, изображенный на листе бумаги. Если граф планарен и нарисован так, что никакие линии не пересекаются, и его необходимо разрезать вдоль ребер, то граф окажется разделенным на части, включая внешнюю часть. Такие части называются *гранями*. Граница каждой грани является циклом.

- Грань планарного графа – максимальный участок плоскости такой, что любые две точки этого участка могут быть соединены кривой, не пересекающей ребро графа.

**ТЕОРЕМА 1.7.** Если  $G$  – связный планарный граф, содержащий  $v$  вершин,  $e$  ребер и  $f$  граней, то  $v - e + f = 2$ .

**ТЕОРЕМА 1.8.** Полный двудольный граф  $K_{3,3}$  не является планарным.

**Лемма.** В произвольном связном планарном графе  $G$  с количеством вершин не менее трех имеет место неравенство  $3v - e \geq 6$ .

**ТЕОРЕМА 1.9.** Полный граф  $K_5$  не является планарным.

**ТЕОРЕМА 1.10.** Каждый планарный граф  $G$  содержит вершину степени 5 или менее.

**Пример.** Покажем, что граф, изображенный на рис. 1.36, является, планарным. Передвигая вершину  $d$ , получаем более простой граф, изображенный на рис. 1.37. Передвигая вершину  $c$ , получаем граф (рис. 1.38), который, очевидно, является планарным.

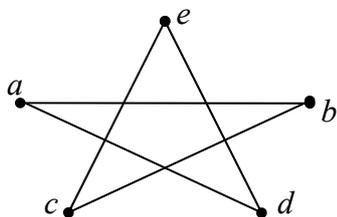


Рис. 1.36

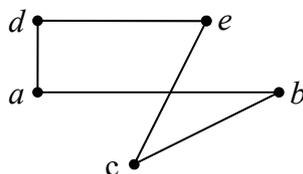


Рис. 1.37

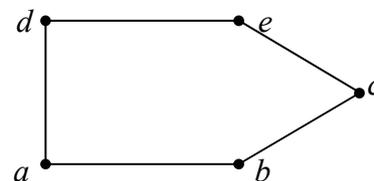


Рис. 1.38

## 1.5 Пути и циклы Гамильтона

В 1857 году математик Уильям Фоуэн Гамильтон придумал игру. Она включала додекаэдр, т.е. правильным многогранник, 12 граней которого представляли собой конгруэнтные правильные пятиугольники. В каждом из 20 углов, или вершин тела, просверливалась дырка, в которую вставлялся колышек, изображавший город. Используя веревку, требовалось найти путь через города, посетив каждый город один раз, и вернуться в исходный город. Додекаэдр на плоскости изображается так, как показано на рис. 1.39.

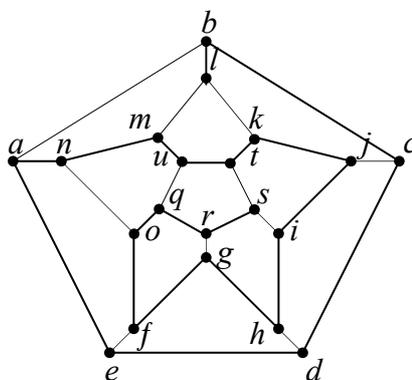


Рис. 1.39

Проблема в таком случае сводится к нахождению цикла в графе, проходящего через каждую вершину, исключая начальную, только один раз. Отсюда любой цикл графа, обладающий таким свойством, называется *гамильтоновым* циклом. Этот цикл в некотором смысле противоположен эйлерову циклу, который проходит через все ребра только один раз.

• Пусть  $G$  – граф. *Гамильтонов путь* – это простой путь, который проходит через каждую вершину  $G$ .

• *Гамильтонов цикл* – это простой цикл, который проходит через каждую вершину графа  $G$ .

Легко убедиться, что граф для игры Гамильтона действительно имеет гамильтонов цикл. Один из таких циклов изображен на рис. 1.39.

**Пример.** Граф на рис. 1.40 имеет гамильтонов цикл, изображенный на рис. 1.41.

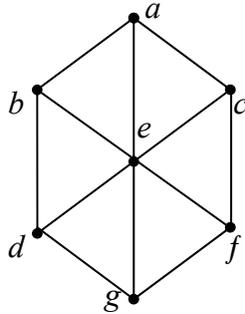


Рис. 1.40

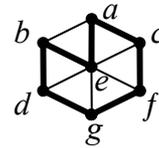


Рис. 1.41

**ТЕОРЕМА 1.11.** Для любой вершины из цикла Гамильтона существует ровно два ребра из этого цикла, инцидентные данной вершине.

**Доказательство.** По ходу цикла для каждой вершины  $V$  имеется ребро к циклу и ребро из цикла. Если бы существовало еще одно ребро цикла инцидентное вершине  $V$ , то цикл вернулся бы в вершину  $V$ , и она опять появилась бы в цикле, что противоречит определению гамильтонова цикла. Следовательно, существует ровно два ребра, которые инцидентны вершине  $V$  из цикла Гамильтона.

Заметим, что граф, у которого есть гамильтонов цикл, называют *гамильтоновым графом*.

**Пример.** Граф Петерсена, изображенный на рис. 1.42 имеет гамильтонов путь, но не имеет гамильтонова цикла. Путь показан на рис. 1.43.

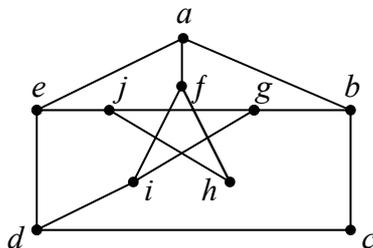


Рис. 1.42

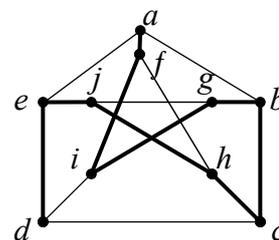


Рис. 1.43

**ТЕОРЕМА 1.12.** Если  $G = G(V, E)$  – связный граф с  $n$  вершинами, где  $n \geq 3$ , и для каждой пары различных несмежных вершин  $u, v \in V$ ,  $\deg(u) + \deg(v) \geq n$ , тогда граф  $G$  имеет гамильтонов цикл.

**Следствие.** Если  $G = G(V, E)$  – связный граф, имеющий  $n$  вершин, где  $n \geq 3$ , и если для каждой вершины  $v \in V$  выполняется  $\deg(v) \geq \frac{n}{2}$ , то граф  $G$  имеет гамильтонов цикл.

**ТЕОРЕМА 1.13.** Пусть  $G = G(V, E)$  – связный граф с  $n \geq 3$  вершинами и пусть  $u$  и  $v$  – несмежные вершины графа  $G$  такие, что  $\deg(u) + \deg(v) \geq n$ . Отсюда граф  $G^e$ , состоящий из графа  $G$  с присоединенным ребром  $e = \{u, v\}$ , имеет гамильтонов цикл тогда и только тогда, когда граф  $G$  имеет гамильтонов цикл.

• Пусть  $G$  – граф с  $n$  вершинами. *Замыканием* графа  $G$ , обозначаемым  $cl(G)$ , называется граф, полученный из графа  $G$  рекурсивным добавлением ребер к несмежным вершинам  $u$  и  $v$  графа  $G$ , для которых  $\deg(u) + \deg(v) \geq n$  до тех пор, пока это возможно.

**Пример.** Если  $G$  – граф, изображенный на рис. 1.44, то  $cl(G)$  – граф, изображенный на рис. 1.45.

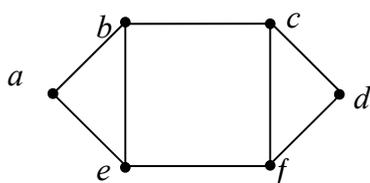


Рис. 1.44

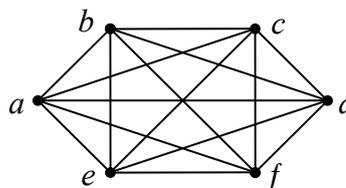


Рис. 1.45

**Пример.** Если  $G$  – граф, изображенный на рис. 1.46, то  $cl(G)$  – граф, изображенный на рис. 1.47.

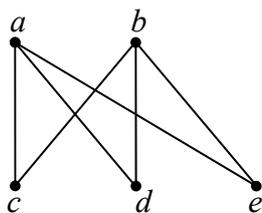


Рис. 1.46

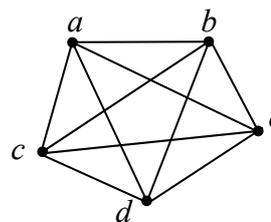


Рис. 1.47

**ТЕОРЕМА 1.14.** Граф  $G$  имеет гамильтонов цикл тогда и только тогда, когда граф  $cl(G)$  имеет гамильтонов цикл.

## 1.6 Алгоритм поиска кратчайшего пути

До сих пор при рассмотрении графов нас интересовали вершины и ребра, по которым можно перемещаться. Теперь нас интересует не только перемещение из точки  $A$  в точку  $B$ , но и то, как это сделать наилучшим способом. Первый вопрос состоит в том, что означает «наилучшим способом». Это может

быть самый дешевый путь, самый безопасный путь, кратчайший путь или тот, который требует минимум энергии, или путь, выбранный в соответствии с каким-то иным критерием. Для определения наилучшего пути присвоим каждому ребру вес или меру. Если пытаться найти кратчайшее расстояние между двумя городами, то их необходимо предоставить в виде вершин, а вес, присвоенный ребрам – это расстояние между городами. Если пытаться найти самый дешевый способ перелета из одного города в другой, то вес ребра между вершинами будет стоимостью перелета из города в город. Если прямого перелета между городами нет, то не будет ребра между соответствующими вершинами. Хотя вес или мера, присвоенные ребрам, могут иметь различные значения, для упрощения будем рассматривать вес ребра как расстояние, а наилучший путь из точки  $A$  в точку  $B$  как кратчайший путь между точками  $A$  и  $B$ . Предполагаем, что вес или мера, названные теперь расстоянием, и приписываемые ребрам между двумя различными точками, положительные.

В дальнейшем будем использовать символ  $\infty$ . Для упрощения рассмотрения предположим, что все целые числа меньше  $\infty$ , так что  $\min(a, \infty) = a$  для каждого неотрицательного целого числа, а  $\min(\infty, \infty) = \infty$ . Примем также, что  $a + \infty = \infty + \infty = \infty$ . Это для удобства обозначений.

- Пусть  $A = (a_1, a_2, a_3, \dots, a_n)$  и  $B = (b_1, b_2, b_3, \dots, b_n)$  – матрицы – строки, где каждое из  $a_i$  и  $b_i$  – неотрицательные целые числа или  $\infty$ . Тогда

$$A \wedge B = (\min(a_1, b_1), \min(a_2, b_2), \min(a_3, b_3), \dots, \min(a_n, b_n))$$

- Пусть  $c$  – число, а  $A = (a_1, a_2, a_3, \dots, a_n)$  – матрица – строка. Тогда

$$c + A = c + (a_1, a_2, a_3, \dots, a_n) = (c + a_1, c + a_2, c + a_3, \dots, c + a_n)$$

Рассмотрим алгоритм Дейкстры. Этот алгоритм позволяет определить не только длину кратчайшего пути, но и сам путь. Это достигается с помощью указателя, который для каждой вершины из кратчайшего пути указывает предыдущую вершину пути. Таким образом, если найдена длина кратчайшего пути между  $A$  и  $B$ , то двигаясь вдоль кратчайшего пути в обратном направлении от  $B$  к  $A$  можно найти сам путь.

**ТЕОРЕМА 1.15.** Пусть  $a = v_1$  и  $b = v_n$ . Если  $v_1, v_2, \dots, v_i, v_{i+1}, \dots, v_j, v_{j+1}, \dots, v_n$  есть кратчайший путь между  $a$  и  $b$ , то  $v_i, v_{i+1}, \dots, v_j$  – часть этого пути между вершинами  $v_i$  и  $v_j$  является кратчайшим путем между  $v_i$  и  $v_j$ .

Начнем с формулировки алгоритма Дейкстры, а затем рассмотрим примеры его использования. Согласно алгоритму отыскивается кратчайшее расстояние от вершины  $v_1$  к вершине  $v_n$ . Начинаем с вершины  $v_1$  и находим расстояние от  $v_1$  до каждой из смежных с ней вершин. Выбираем вершину, расстояние от которой до вершины  $v_1$  наименьшее, пусть это будет вершина  $v_i$ . да-

лее находим расстояние от вершины  $v_i$  до каждой вершины смежной к  $v_i$  вдоль пути, проходящего через вершину  $v_i$ . Если это расстояние меньше, чем текущее расстояние, присвоенное каждой из вершин, то заменяем им текущее расстояние. Снова выбираем вершину, ближайшую к  $v_i$ , но не совпадающую с выбранной ранее, и процесс повторяется.

### Алгоритм Дейкстры

Для данного взвешенного графа алгоритм дает кратчайшее расстояние от вершины  $v_1$  к вершине  $v_n$ . Каждой вершине поставим в соответствие упорядоченную пару  $(\infty; 0)$ . Первая координата вершины  $v_i(m, v_r)$  будет означать присвоенное расстояние от вершины  $v_1$  к вершине  $v_i$ , а вторая координата – предыдущую вершину пути от  $v_1$  к  $v_i$ .

1. Начать в вершине  $v_1(\infty; 0)$ , заменить ее на  $v_1(0, 0)$  и сделать постоянной. Остальные вершины на этот момент оставить временными.

2. Когда вершина  $v_k(m, v_r)$  станет постоянной, для каждой вершины  $v_j$  смежной к  $v_k$  прибавить величину  $m$  к расстоянию от вершины  $v_k$  к вершине  $v_j$ . Если это значение меньше, чем текущее расстояние, присвоенное вершине  $v_j$ , заменить текущее расстояние этой суммой и заменить вторую координату на  $v_k$ .

3. Найти минимум из расстояний, приписанных временным вершинам. Первую из вершин с таким расстоянием делаем постоянной.

4. Если  $v_n$  – не постоянная вершина, то возвращаемся к п.2.

5. Если  $v_n$  – постоянная вершина, то расстояние, присвоенное вершине  $v_n$  является кратчайшим расстоянием от  $v_1$  к  $v_n$ .

6. Для нахождения пути начать в вершине  $v_n$ , найти предшествующую ей вершину пути (вторая координата). Для каждой вершины пути  $v_j$  находить предшествующую ей вершину пути, пока не будет достигнута вершина  $v_1$ . Перестановка вершин в обратном порядке даст кратчайший путь.

**Пример.** Пусть граф изображенный на рисунке 1.48 – взвешенный граф, в котором отыскивается кратчайшее расстояние от вершины  $A$  к вершине  $F$ . Поставив в соответствие каждой вершине упорядоченную пару  $(\infty; 0)$  рассматриваемый граф приводим к виду, показанному на рис. 1.49.

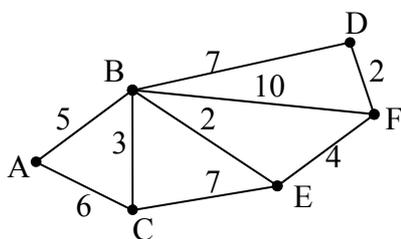


Рис. 1.48

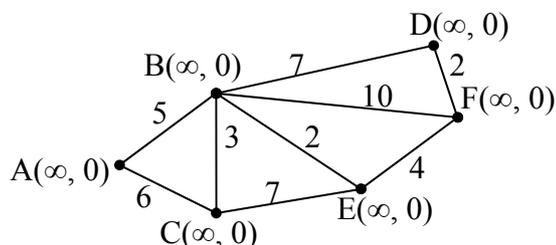


Рис. 1.49

Приступим к построению путей от вершины  $A$  к другим вершинам. Первая компонента упорядоченной пары покажет длину кратчайшего пути к вер-

шине в момент достижения, а вторая компонента укажет на предыдущую вершину кратчайшего пути. Первая компонента будет содержать  $\infty$ , а вторая – 0 до тех пор, пока путь не найден. Вершина, которая стала постоянной, будет выделена. Выполнив шаг 1 алгоритма, получаем граф, изображенный на рисунке 1.50.

Поскольку вершины  $B$  и  $C$  – смежные с вершиной  $A$ , выполняем шаг 2 и упорядоченной паре для вершины  $B$  присваиваем значение  $(5, A)$ , а упорядоченной паре для вершины  $C$  присваиваем значение  $(6, A)$ . (Фактически изменения вносятся, тогда и только тогда, когда новые расстояния меньше старых, но поскольку старые расстояния до вершины  $B$  и  $C$  равны  $\infty$ , в данном случае это не имеет значения). Выполнив шаг 3, выбираем наименьшее из временных присвоенных значений. В данном случае это расстояние до вершины  $A$  равно 5, и вершину  $B(5, A)$  делаем постоянной. Таким образом, получаем рисунок 1.51.

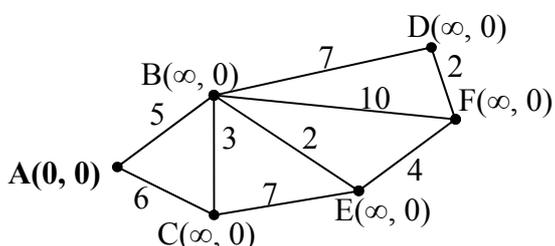


Рис. 1.50

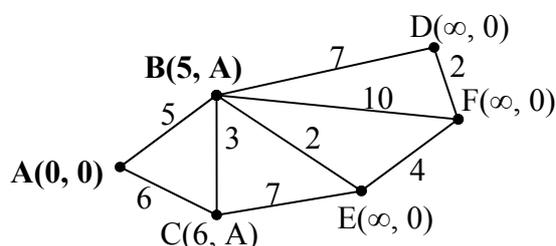


Рис. 1.51

Возвращаясь к шагу 2, рассмотрим временные вершины  $C, D, E, F$ , смежные с вершиной  $B$ . В каждом случае прибавляем расстояние от вершины  $A$  к вершине  $B$  к расстоянию от вершины  $B$  к данным вершинам. Таким образом, для вершины  $C$  это будет  $5 + 3 = 8$ . Для вершины  $D$  имеем  $5 + 7 = 12$ . Для вершины  $E$  имеем  $5 + 2 = 7$ . Для вершины  $F$  имеем  $5 + 10 = 15$ . Поскольку новое расстояние до вершины  $C$  не меньше, чем уже присвоенное, упорядоченную пару  $C(6, A)$  оставляем без изменений. Новые расстояния до вершин  $D, E, F$  меньше уже присвоенных, поэтому им задаем значения, которые получены для пути из вершины  $B$ , то есть меняем их на  $D(12, B), E(7, B), F(15, B)$ .

Выполнив шаг 3, находим наименьшее из расстояний, присвоенных временным вершинам, поэтому берем  $\min\{6, 12, 15, 7\} = 6$ , и поскольку вершина  $C$  имеет это расстояние делаем вершину  $C(6, A)$  постоянной. Таким образом, получаем рис. 1.52.

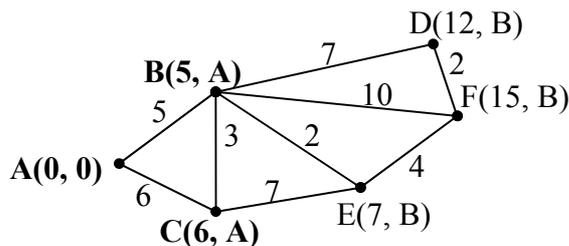


Рис. 1.52

Теперь берем новую постоянную вершину  $C$ . Выполнение шага 2 не приводит к изменениям. Выполнив шаг 3, делаем вершину  $E(7, B)$  постоянной. Получает в результате рис. 1.53.

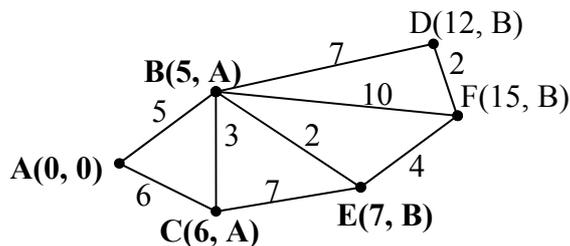


Рис. 1.53

Берем новую постоянную вершину  $E$  и, используя шаг 2, меняем  $F(15, B)$  на  $F(11, E)$ . Выполнив шаг 3, делаем вершину  $F(11, E)$  постоянной. Таким образом, получаем рис. 1.54.

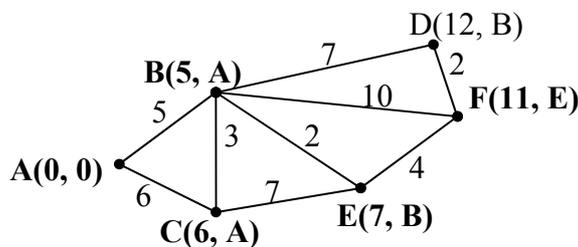


Рис. 1.54

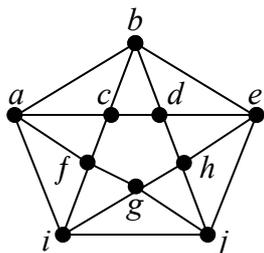
Вершина  $F$  стала постоянной, поэтому процесс завершен и 11 – это кратчайшее расстояние от вершины  $A$  к вершине  $F$ . Если бы совокупность вершин смежных с постоянной вершиной, была исчерпана до того, как мы достигли вершину  $F$ , то задача не имела бы решения, поскольку бы не было бы пути от вершины  $A$  к вершине  $F$ . Для нахождения кратчайшего пути заметим, что вершине  $F$  предшествует вершина  $E$ , вершине  $E$  предшествует вершина  $B$ , а вершине  $B$  предшествует вершина  $A$ . Поэтому кратчайшим путем является  $ABEF$ .

## УПРАЖНЕНИЯ

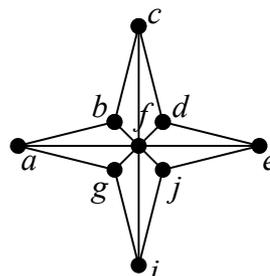
I. Среди приведенных ниже графов найдите те, которые имеют эйлеров цикл.

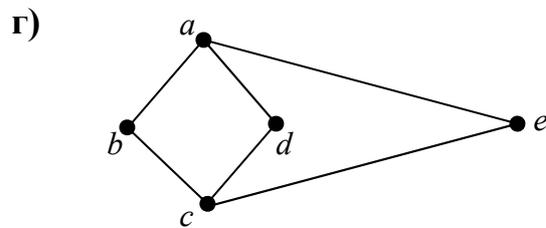
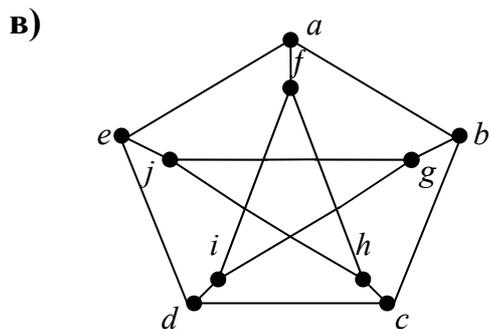
1.

a)

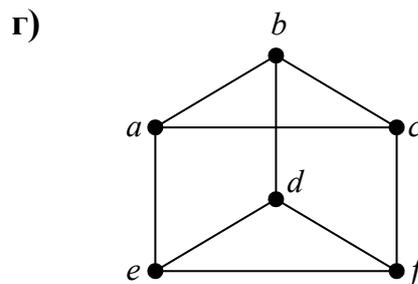
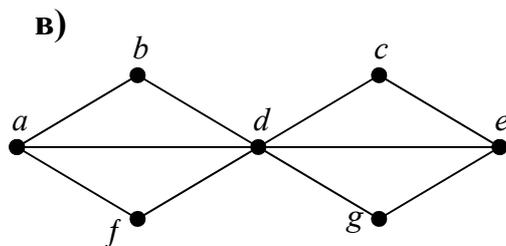
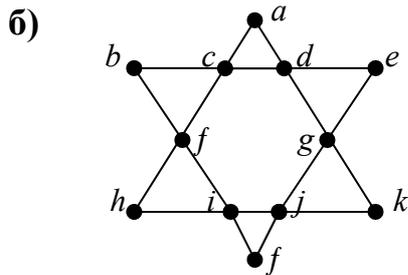
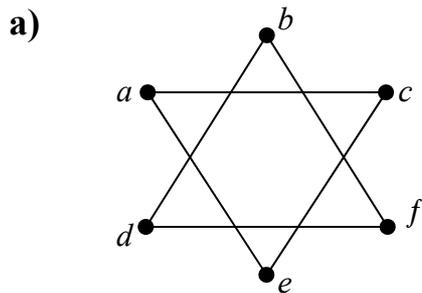


б)

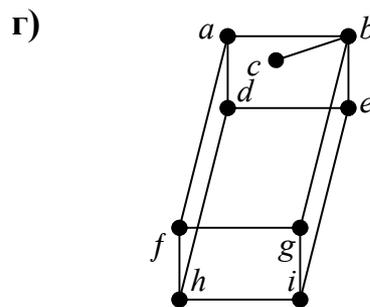
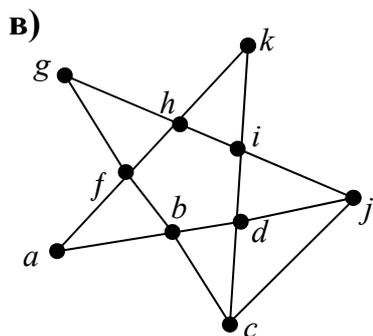
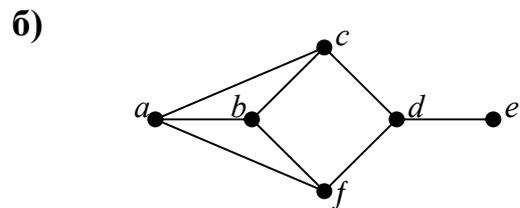
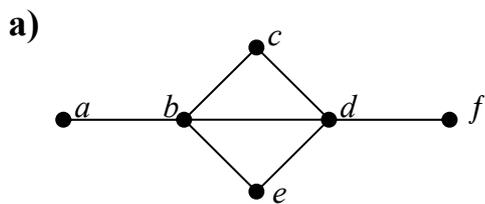




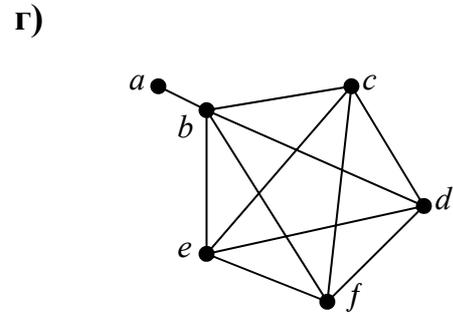
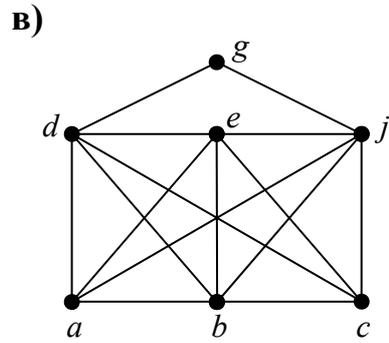
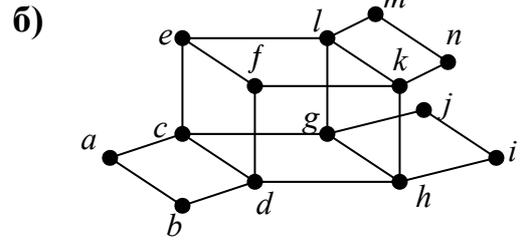
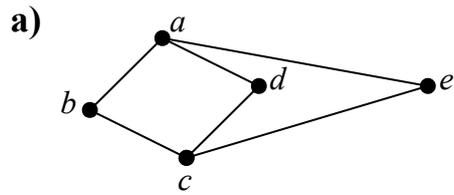
2.



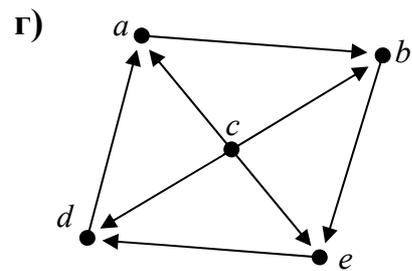
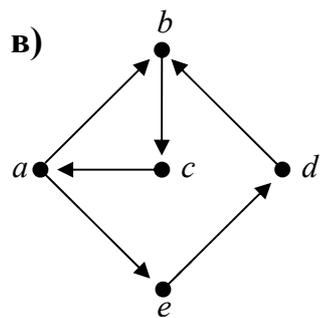
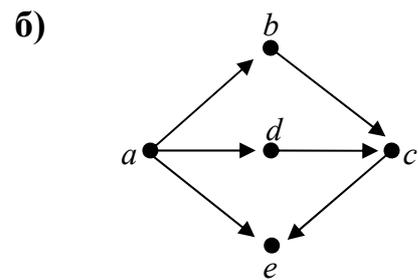
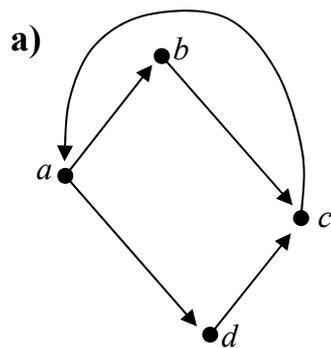
3. Среди приведенных ниже графов найдите те, которые имеют собственный эйлеров цикл.



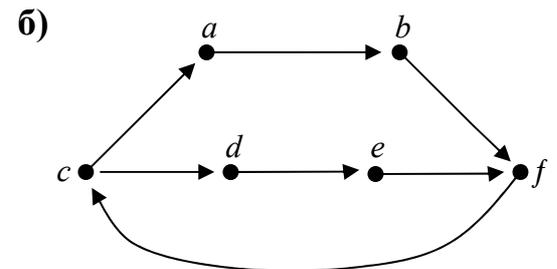
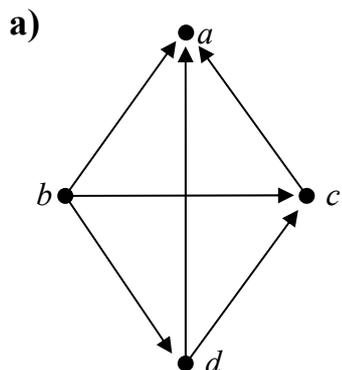
4.

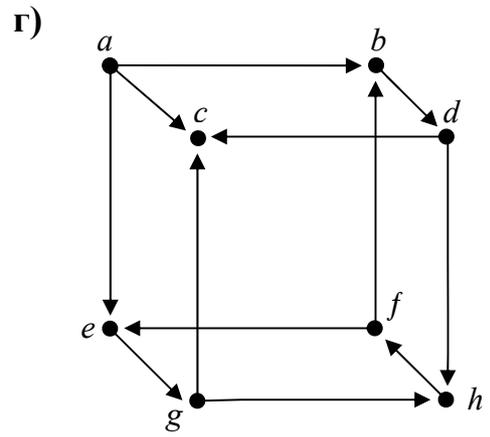
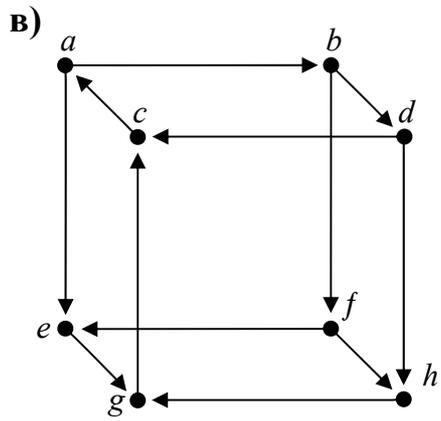


5. Какие из следующих графов являются сильно связными?

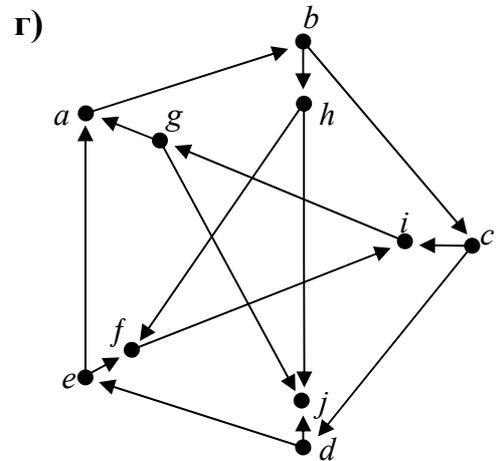
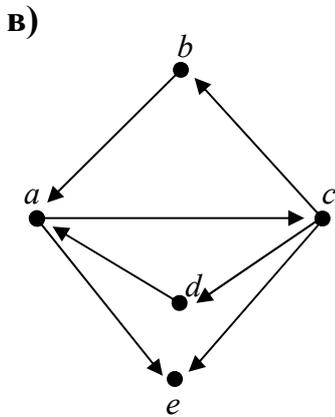
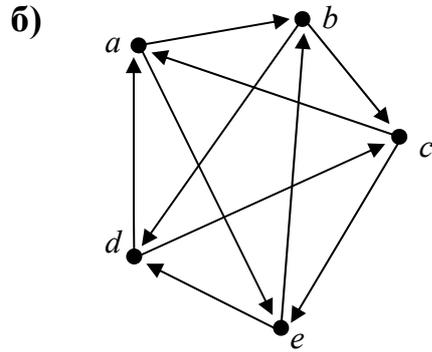
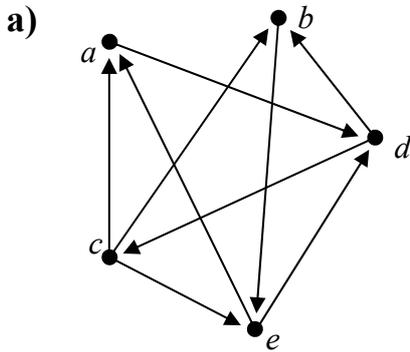


6.

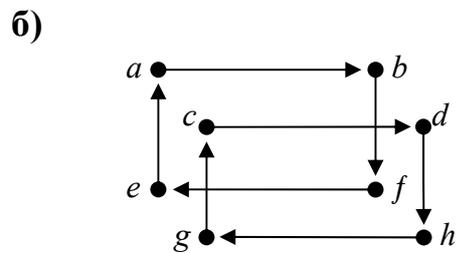
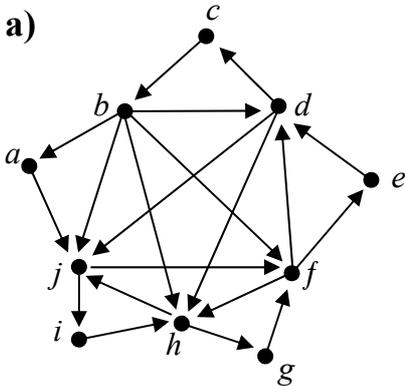


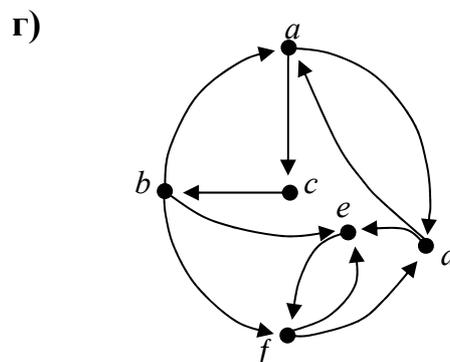
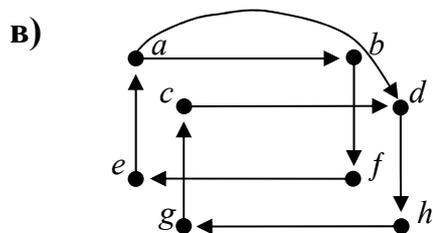


7. Какие из следующих ориентированных графов имеют эйлеровы циклы?



8.





9. Докажите теорему. Граф (мультиграф или псевдограф) имеет собственный эйлеров путь тогда и только тогда, когда он связный и ровно две вершины имеют нечетную степень.

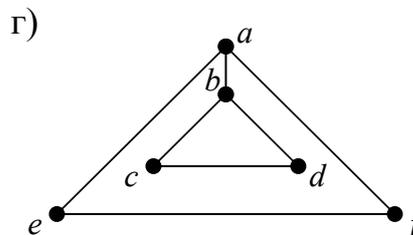
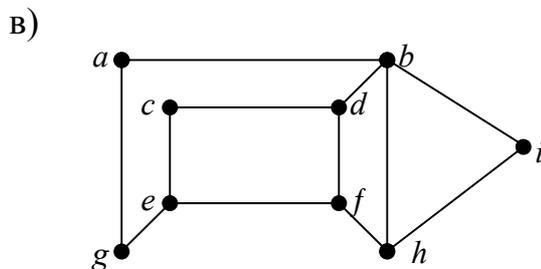
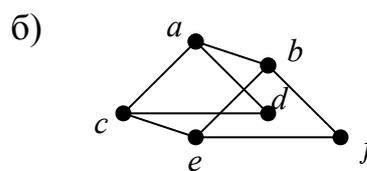
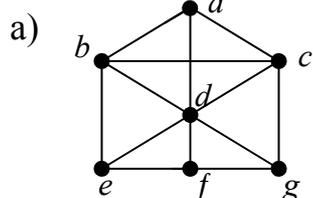
10. Докажите, что если граф содержит цикл от вершины  $v$  к ней самой, то содержит простой цикл от вершины  $v$  к ней самой.

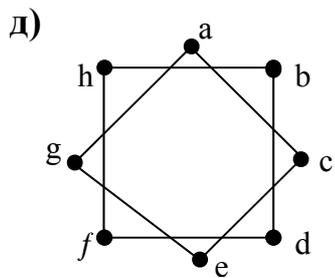
11. Докажите, что ориентированный граф является сильно связным, если в графе существует вершина  $v$  такая, что каждая другая вершина графа достижима из  $v$  и вершина  $v$  достижима из любой другой вершины графа.

12. Докажите теорему. Ориентированный граф имеет эйлеров путь тогда и только тогда, когда он сильно связный и степень входа каждой вершины равна ее степени выхода.

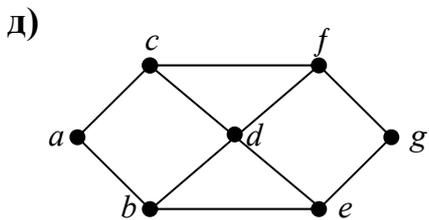
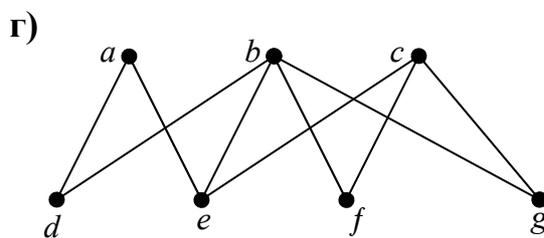
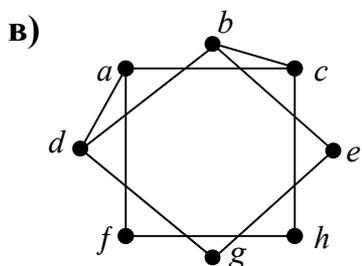
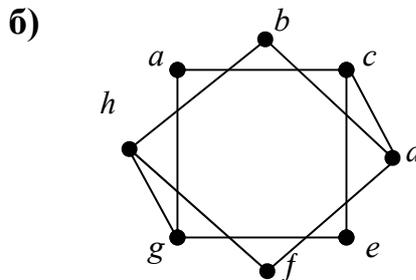
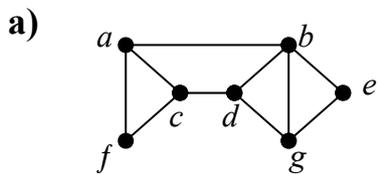
II. Найдите гамильтонов цикл, если он существует, для каждого из приведенных ниже графов.

13.

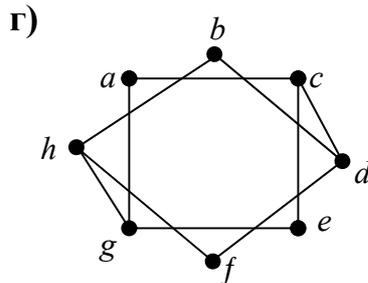
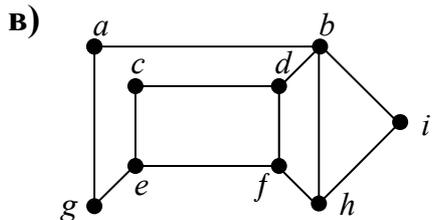
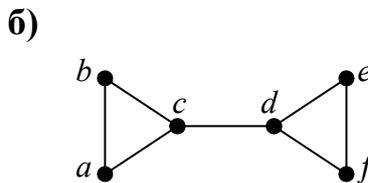
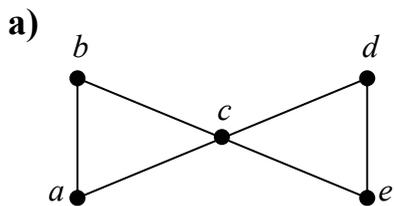


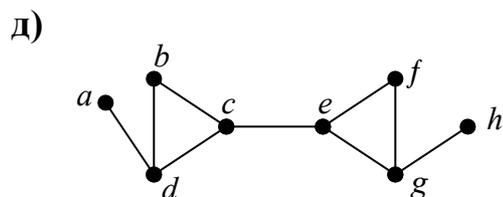


14.

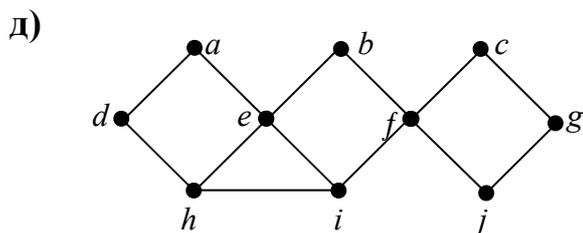
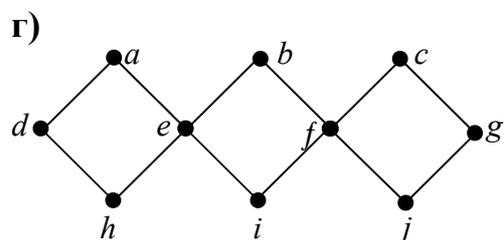
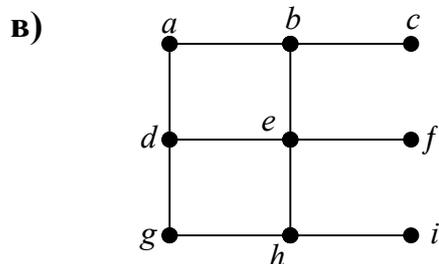
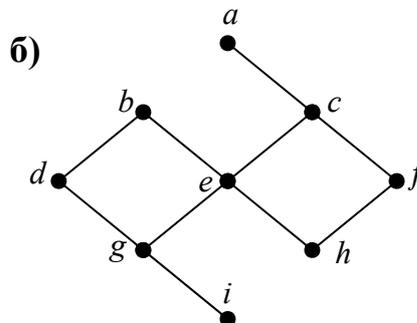
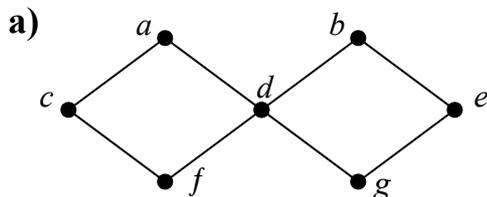


15.





16.



17. Докажите теорему. Если граф  $G$  имеет разрезающее ребро, то граф  $G$  не может иметь гамильтонов цикл. Если компоненты графа, полученные при удалении разрезающего ребра, имеют гамильтоновы циклы, то граф  $G$  имеет гамильтонов путь.

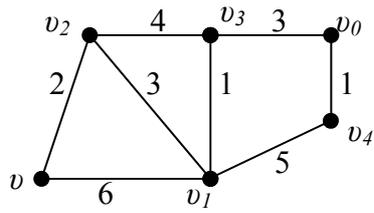
18. Нарисуйте граф с шестью вершинами, который имеет гамильтонов цикл, но не имеет эйлерова цикла.

19. Нарисуйте граф с шестью вершинами, который имеет эйлеров цикл, но не имеет гамильтонова цикла.

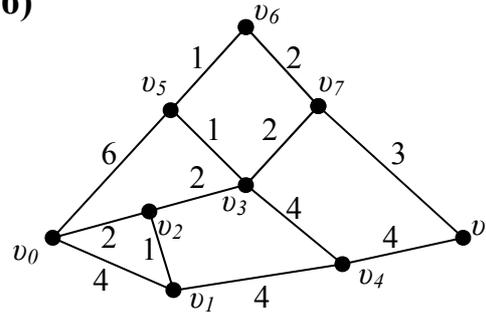
III. Используйте алгоритм Дейкстры для нахождения в приведенных ниже графах, кратчайшего расстояния между вершинами  $v_0$  и  $v$ .

20.

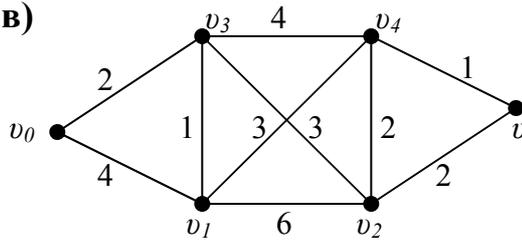
a)



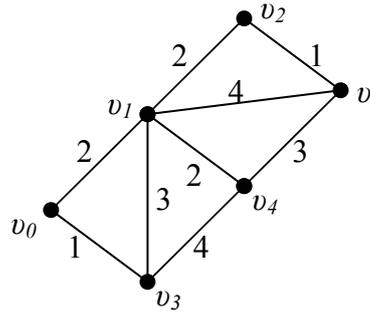
б)



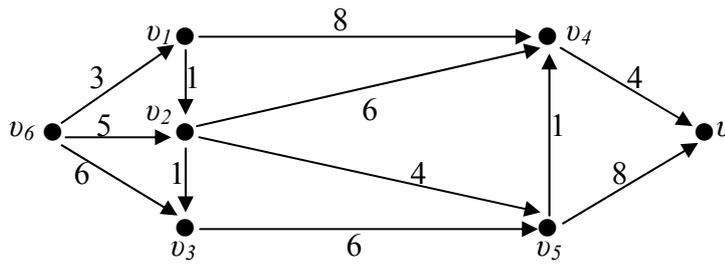
в)



г)

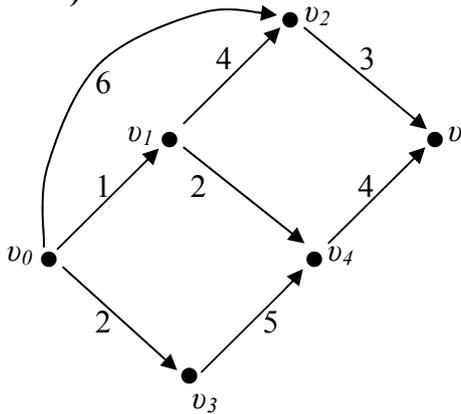


г)

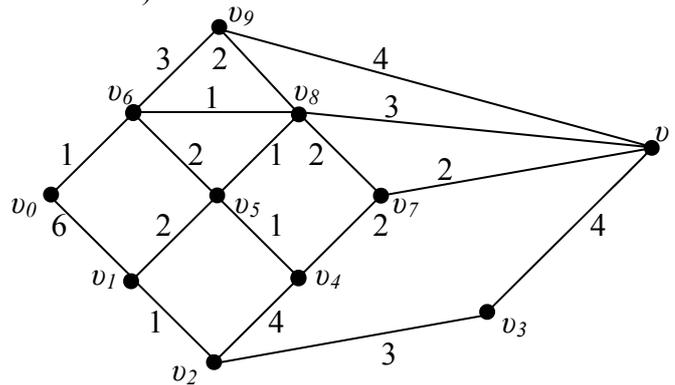


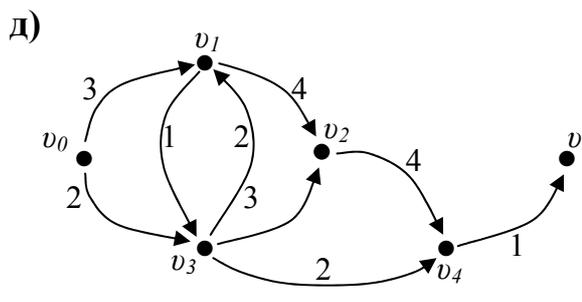
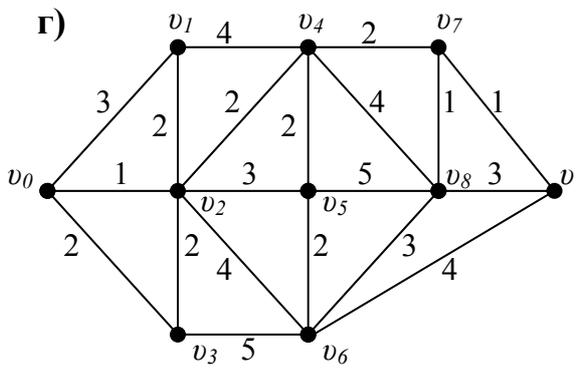
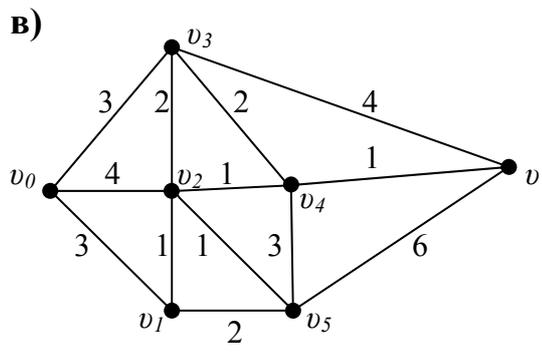
21.

a)



б)





## 2. СЕТИ

### 2. 1. Сети и потоки

Сеть можно представить как систему, транспортирующую некий продукт из одной точки в другую. Этим продуктом могут быть люди, электроэнергия, природный газ, нефть и многое другое. Примером может служить система нефтепровода, где нефть течет из одной точки к другим точкам системы. Наша терминология будет соответствовать данной концепции. Используя такое представление, рассмотрим сеть как ориентированный граф, ребра которого – трубы между точками системы, а они, в свою очередь, представлены вершинами графа. Каждому ребру  $e = (v_i, v_j)$  соответствует положительное число  $c(e)$ , называемое пропускной способностью  $e$ . Если между двумя вершинами не существует ребра, то пропускную способность полагаем равной нулю. В примере с нефтяными сетями пропускная способность связана с количеством нефти, проходящей через трубу (ребро).

Перед тем как определить сеть, ограничим определение ориентированного графа. Наличие петель у графа недопустимо, поскольку рассматриваемые задачи связаны только с транспортировкой продукта между различными точками. Будем считать, что если существует ребро из  $v_i$  в  $v_j$ , то нет ребра из  $v_j$  в  $v_i$ . Таким образом, рассматривается поток продукта только в одну сторону. Необходимое требование: ориентированный граф должен быть связан, т.к. если имеется путь из  $a$  в  $z$ , нас будет интересовать только компонента, содержащая  $a$  и  $z$ . Если между  $a$  и  $z$  не существует пути, то и определять нечего. Ориентированный граф, удовлетворяющий этим условиям, часто называется простым связным ориентированным графом. Назовем такой граф ориентированным графом, имея в виду, что он удовлетворяет упомянутым выше ограничениям. Рассмотрим также особую вершину  $a$ , называемую источником, и особую вершину  $z$ , называемую стоком. Степень входа вершины  $a$  равна 0, так что в источник ничто не втекает. Степень выхода вершины  $z$  равна 0, так что из стока ничто не вытекает. Таким образом, продукт перевозится из  $a$  и имеет место назначения  $z$ . Более точно определим сеть следующим образом.

• *Сеть* – это ориентированный граф  $(G, V, E)$  вместе с весовой функцией  $C : E \rightarrow N$  и выделенными вершинами  $a, z$ , такими что

$$(i) \text{integ}(a) = 0;$$

$$(ii) \text{integ}(z) = 0.$$

**Например,** граф, изображенный на рис. 2.1, является примером сети, где числа на каждом ребре обозначают пропускную способность.

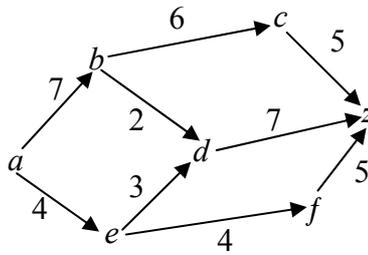


Рис. 2.1

Для этой сети, которая представлена как нефтепровод, введем понятие потока (количество нефти, проходящее через трубы нефтепровода). Таким образом, для каждого ребра  $e$  имеется значение  $f(e)$ , которое является потоком через конкретное ребро или трубу. Очевидно, величина потока не может превысить пропускную способность трубы. Потребуем также, чтобы поток, входящий в вершину, был равен потоку, выходящему из вершины, за исключением вершин  $a$  и  $z$ .

Это называется *сохранением потока*. Пусть  $in(v)$  – множество ребер, для которых  $v$  – конечная вершина, и  $out(v)$  – множество ребер, для которых  $v$  – начальная вершина. Таким образом,  $out(v)$  – множество ребер, выходящих из вершины  $v$ , и  $in(v)$  – множество ребер, входящих в вершину  $v$ . Следовательно, имеем следующее определение.

• Поток в сети – это функция  $f : E \rightarrow N \cup \{0\}$  такая, что

(i) для всех  $e \in E, 0 \leq f(e) \leq c(e)$ ;

(ii) для всех  $v \in V$  таких, что  $v \neq a, z, \sum_{e \in in(v)} f(e) = \sum_{e \in out(v)} f(e)$ .

Допустим, имеется фиксированный поток. Пусть  $поток(a) = \sum_{e \in out(a)} f(e)$ , т.е.  $поток(a)$  – поток, вытекающий из источника  $a$ , и  $поток(z) = \sum_{e \in in(z)} f(e)$ , т.е.  $поток(z)$  – поток, втекающий в сток  $z$ . Пример потока в сети, где первый элемент упорядоченной пары на каждом ребре – пропускная способность, а второй – поток, демонстрирует рис. 2.2.

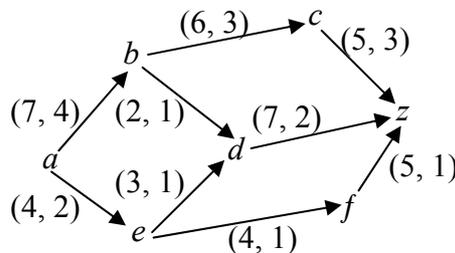


Рис. 2.2

Поток через каждое ребро меньше, чем его пропускная способность. Обратите внимание, что для вершины  $b$  поток, втекающий в  $b$ , равен 4 и поток,

вытекающий из  $b$ , равен 4. Таким образом, оба потока совпадают, и в вершине  $b$  имеет место сохранение потока. Это верно также и для всех других вершин, за исключением вершин  $a$  и  $z$ . Воспользуемся принципом сохранения потока для доказательства того, что интуитивно кажется очевидным. А именно, поток из  $a$  равен потоку в  $z$ , т.е.  $\text{поток}(a) = \text{поток}(z)$ , что и утверждает приведенная ниже теорема. Рассмотрим сначала конкретную сеть.

Заметим, что для потока на рис 2.2 выполняется равенство  $\text{поток}(a) = \text{поток}(z) = 6$ . Пусть  $S$  – множество вершин  $\{b, c, d\}$ , а  $T = V - S$  – множество вершин  $\{a, f, z\}$ . Если просуммировать потоки в  $S$ , то согласно приведенной ниже таблице 2.1 получим, что входящий поток равен 9 и выходящий поток равен 9. Таким образом, разность потоков равна 0.

Таблица 2.1

вершина	входящий поток	ребро	выходящий поток	ребро
b	4	(a, b)	3	(b, c)
			1	(b, d)
c	3	(b, c)	3	(c, z)
d	1	(e, d)	2	(d, z)
	1	(b, d)		
сумма	9		9	

Если во множество  $S$  добавить вершину  $a$ , получим, что разность 15 (выходящий поток) и 9 (входящий поток) равна  $6 = \text{поток}(a)$ .

Таблица 2.2

вершина	входящий поток	ребро	выходящий поток	ребро
b	4	(a, b)	3	(b, c)
			1	(b, d)
c	3	(b, c)	3	(c, z)
d	1	(e, d)	2	(d, z)
	1	(b, d)		
a	0		4	(a, b)
			2	(a, e)
сумма	9		15	

Заметим, что ребра  $(a, b)$ ,  $(b, c)$  и  $(b, d)$  присутствуют в колонке для входящего и выходящего потока, потому что обе вершины каждого ребра принадлежат  $S$ . При вычитании эти ребра сокращаются, поэтому, удалив их из обеих частей, получаем

Таблица 2.3

вершина	входящий поток	ребро	выходящий поток	ребро
b				
c			3	(c, z)
d	1	(e, d)	2	(d, z)

вершина	входящий поток	ребро	выходящий поток	ребро
а	0			
			2	(а, е)
сумма	1		7	

Снова разница входящего и выходящего потока равна 6. Но поскольку ребра, обе вершины которых принадлежат  $S$ , удалены, то выходящий поток – это сумма потоков ребер, которые идут из  $S$  в  $T$ , а входящий поток – это сумма потоков ребер, которые идут из  $T$  в  $S$ . Аналогичные рассуждения будут использованы в приведенной ниже теореме.

**ТЕОРЕМА 2.1.** Для любого фиксированного потока  $f$

$$\text{поток}(a) = \sum_{e \in \text{out}(a)} f(e) = \sum_{e \in \text{in}(z)} f(e) = \text{поток}(z).$$

**СЛЕДСТВИЕ.** Пусть  $S$  – подмножество множества  $V$ , содержащее  $a$ , но не содержащее  $z$ , и  $T = V - S$ . Тогда

$$\sum_{e \in (S,T)} f(e) - \sum_{e \in (T,S)} f(e) = \text{поток}(a) = \text{поток}(z).$$

- Величина потока  $f$ , обозначаемая как  $val(f)$ , равна  $\text{поток}(a) = \text{поток}(z)$ .
- Пусть  $S$  – подмножество множества  $V$  и  $T = V - S$ . Тогда  $\{e : e \in (S,T)\}$  называется *сечением*. Если  $a \in S$  и  $z \in T$ , то сечение называется  *$a - z$  сечением*.
- Величина  $C(S,T) = \sum_{e \in (S,T)} c(e)$  называется *пропускной способностью сечения*.
- Поток  $f$  в сети называется *максимальным потоком*, если  $val(f) \geq val(f)$  для любого возможного потока  $f$  в сети.
- $a - z$  сечение  $(S, T)$  называется *минимальным сечением*, если  $C(S, T)$  не больше пропускной способности любого другого  $a - z$  сечения.

**ТЕОРЕМА 2.2.** Пусть  $S$  – подмножество множества  $V$ , содержащее  $a$ , но не содержащее  $z$ , и  $T = V - S$ . Тогда

$$val(f) \leq C(S, T).$$

**СЛЕДСТВИЕ 2.2.1.** Если  $val(f) = C(S, T)$  для некоторого потока  $f$ , а  $a - z$  сечения  $(S, T)$ , то  $f$  – максимальный поток, а  $C$  – минимальное сечение.

**СЛЕДСТВИЕ 2.2.2.** Для некоторого потока  $f$  и  $a - z$  сечения  $(S, T)$  равенство  $val(f) = C(S, T)$  имеет место тогда и только тогда, когда  $f(e) = c(e)$  для всех  $e \in (S, T)$  и  $f(e) = 0$  для всех  $e \in (T, S)$ .

Опишем способы определения максимального потока сети. Рассмотрим сеть, изображенную на рис. 2.3.

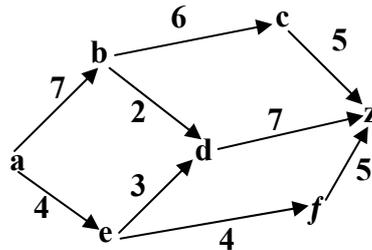


Рис. 2.3

Максимальный поток (не обязательно единственный) легко найти способом, проиллюстрированным на рис. 2.4.

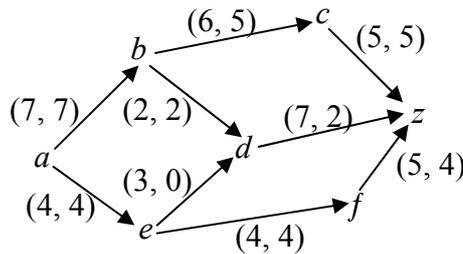


Рис. 2.4.

В данном случае известно, что поток максимальный, потому что величина потока, выходящего из  $a$ , не может превысить сумму пропускных способностей ребер, выходящих из  $a$ .

Рассмотрим сеть, изображенную на рис. 2.5.

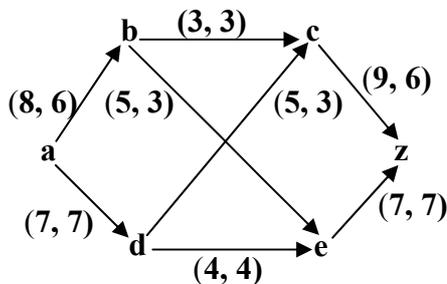


Рис. 2.5.

Кажется, что у этой сети максимальный поток, потому что не существует ориентированного пути, по которому можно увеличить поток. Заметим, однако, что сеть на рис. 2.6 имеет больший поток, и, фактически, он максимальный.

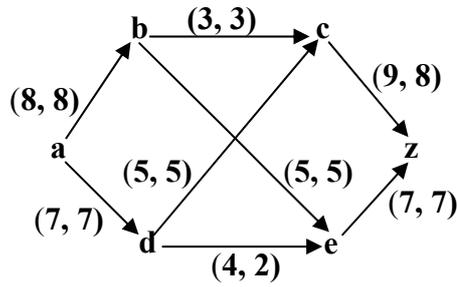


Рис. 2.6

Следует обратить внимание, что если поток из источника равен сумме пропускных способностей ребер, выходящих из источника, или поток, втекающий в сток, равен сумме пропускных способностей ребер, входящих в сток, то поток максимальный. Однако, поток может быть максимальным и без удовлетворения какого-либо из этих признаков. Например, сеть на рис. 2.7 имеет максимальный поток.

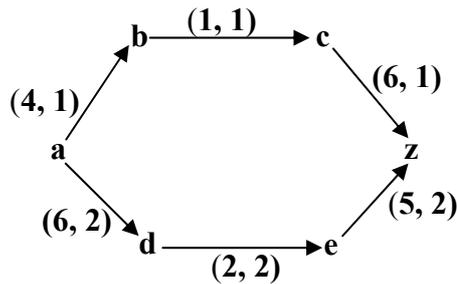


Рис. 2.7

Итак, как же находить максимальные в смысле потока сети? Для этого сформируем пути из  $a$  в  $z$ , не обращая внимания на направление ребер. Такие пути назовем цепями. Рассмотрим снова сеть с потоком, изображенную на рис. 2.5. Одним из таких путей является

$$a \xrightarrow{(8,6)} b \xrightarrow{(3,3)} c \xrightarrow{(9,6)} z.$$

Очевидно, что нет возможности увеличить поток по этому пути, поскольку ребро из  $b$  в  $c$  наполнено до его пропускной способности. То же самое верно и для цепи

$$a \xrightarrow{(8,6)} b \xrightarrow{(5,3)} e \xrightarrow{(7,7)} z.$$

Однако, при рассмотрении цепи

$$a \xrightarrow{(8,6)} b \xrightarrow{(5,3)} e \xrightarrow{(4,4)} d \xrightarrow{(5,3)} c \xrightarrow{(9,6)} z$$

видим: если увеличить поток на 2 для стрелок, имеющих то же самое направление, что и цепь, и уменьшить поток на 2 для стрелок, имеющих противоположное направление, то получим

$$a \xrightarrow{(8,8)} b \xrightarrow{(5,5)} e \xrightarrow{(4,2)} d \xrightarrow{(5,5)} c \xrightarrow{(9,8)} z,$$

что увеличивает поток на 2. Первый вопрос, который следовало бы задать: «Почему выбираем 2?». Видно очевидно, что поток желательно увеличить как можно больше. Но он не может превысить пропускную способность ни одного из заданных ребер. Именно по этой причине мы ограничены величиной 2. Кроме того, если ребро имеет направление, противоположное цепи, то нельзя уменьшить поток через это ребро более, чем на текущую величину потока через него, иначе получим отрицательный поток. Поэтому, если бы не было других ограничений, то

$$e \xleftarrow{(4,4)} d$$

ограничило бы изменение потока до 4. Второй вопрос: «Как это влияет на сохранение потока?». Убедимся, что условие сохранения потока выполняется. Рассмотрим, например, изменение

$$a \xrightarrow{(8,6)} b \xrightarrow{(5,3)} e$$

на

$$a \xrightarrow{(8,8)} b \xrightarrow{(5,5)} e.$$

Поток выходящий из вершины  $b$ , увеличен на ту же самую величину, что и поток, входящий в вершину  $b$ . Поэтому чистый поток через  $b$  остается прежним. При изменении

$$b \xrightarrow{(5,3)} e \xrightarrow{(4,4)} d$$

на

$$b \xrightarrow{(5,5)} e \xrightarrow{(4,2)} d$$

поток из вершины  $b$  в вершину  $e$  увеличен на ту же величину, на которую уменьшен поток из вершины  $d$  в вершину  $e$ , поэтому чистый поток в вершине  $e$  остался тем же. Наконец, рассмотрим изменение

$$e \xleftarrow{(4,4)} d \xrightarrow{(5,3)} c$$

на

$$e \xleftarrow{(4,2)} d \xrightarrow{(5,5)} c$$

Поток из  $d$  в  $e$  уменьшен на ту же величину, на которую увеличен поток из  $d$  в  $c$ . Поэтому чистый поток из вершины  $d$  остался неизменным.

Процесс увеличения потока в сети чрезвычайно прост. Формируем цепь из  $a$  в  $z$ . Если возможно, то увеличиваем поток, определяя наибольшую величину, которую можно добавить к каждому из ребер, имеющих то же самое направление, что и цепь, чтобы поток не превысил пропускную способность, и которую можно вычесть из каждого ребра, имеющего противоположное направление, не создавая отрицательный поток. Поскольку последнее ребро, входящее в  $z$ , имеет то же самое направление, что и цепь, то поток в  $z$  возрастает. Аналогично, ребро, выходящее из  $a$ , имеет то же самое направление, что и цепь, поэтому, как и ожидалось, поток из  $a$  возрастает. Так и должно было быть, потому что  $\text{поток}(a) = \text{поток}(z)$ . Потому что пропускная способность – конечная величина, а общая величина потока увеличивается, неизбежно наступает момент, когда наращивать поток далее нельзя. Когда это происходит, получаем максимальный поток.

До сих пор рассматривался способ увеличения уже существующего потока. А с какого потока начинать? Ответ прост. Следует начинать с нулевого потока, т.е. когда поток через каждое ребро равен 0, а затем наращивать его. Теперь перейдем к изложению систематического алгоритма для нахождения максимального потока. Каждой вершине поставим в соответствие упорядоченную пару. Первый элемент пары – предшествующая вершина в рассматриваемой цепи предназначена для того, чтобы можно было найти обратный путь. Второй элемент пары – резерв, т.е. величина, на которую можно увеличить поток на каждом ребре вдоль пути, если ориентация ребра совпадает с направлением цепи (такую ориентацию назовем правильной), или уменьшить поток, если ребро ориентировано неправильно. Проще говоря, резерв заданной вершины равен максимальной величине, на которую поток вдоль цепи до этой вершины можно увеличить без нарушения самого потока. Рассмотрим также множество  $S$ , содержащее все вершины, не задействованные при построении цепи в  $z$ . Если  $S$  окажется пустым, до того как будет достигнута вершина  $z$ , то больше не будет вершин, которые можно испытать на пути к  $z$ , поэтому нет иного пути в  $z$  и нет более возможности увеличить поток, значит поток максимален.

### Алгоритм (Форда-Фалкерсона) нахождения максимального потока

- (1) Установить предшественника каждой вершины и резерв равными символу – (непомечено). Вершина помечена, когда ее резерв не обозначен

символом  $-$ . Установить резерв вершины  $a$  равным  $\infty$ , с тем чтобы она не ограничивала резерв других вершин. Положить  $S = \{a\}$ .

- (2) Если  $S$  – пустое множество, то поток максимизирован. Если  $S$  не является пустым, то выбираем любой элемент из  $S$  и удаляем его. Полагаем этот элемент равным  $v$ .
- (3) Если  $\omega$  не помечена,  $(v, \omega)$  является ребром и  $f((v, \omega)) < c((v, \omega))$ . Положить резерв вершины  $\omega$  равным минимуму величины  $c((v, \omega)) - f((v, \omega))$  и резерва вершины  $v$ . Установить предшественника вершины  $\omega$  в  $v$ . Если  $\omega \neq z$ , добавить  $\omega$  в  $S$ .
- (4) Если  $\omega$  не помечена,  $(\omega, v)$  является ребром и  $f((v, \omega)) > 0$ . Положить резерв вершины  $\omega$  равным минимуму из  $f((v, \omega))$  и резерва вершины  $v$ . Установить предшественника вершины  $\omega$  в  $v$ . Если  $\omega$  во множество  $S$ .
- (5) Если  $z$  помечена, то, используя функцию предшествования, вернуться в вершину  $a$  и для каждого ребра цепи добавить резерв вершины  $z$  к потоку каждого правильно ориентированного ребра, и вычесть резерв  $z$  из потока каждого неправильно ориентированного ребра. Вернуться к шагу 1.
- (6) Вернуться к шагу 2.

Требуется еще доказать, что алгоритм действительно определяет максимальный поток, но сначала посмотрим, как он работает.

**Пример.** Найдем максимальный поток для сети, изображенной на рис. 2.8.

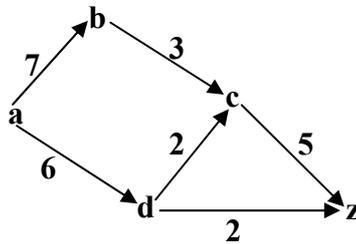


Рис. 2.8

На шаге 1 для каждой вершины устанавливаем предшественника и резерв равными символу  $-$ , для вершины  $a$  устанавливаем резерв, равный  $\infty$ , и полагаем  $S = \{a\}$ . После чего получаем сеть, изображенную на рис. 2.9.

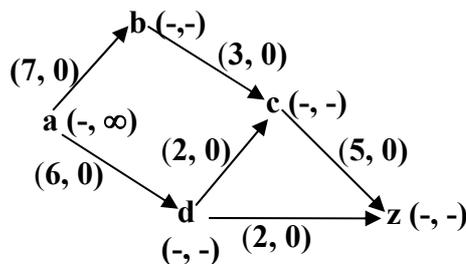


Рис. 2.9

На шаге 2 проверяем, не пусто ли множество  $S$ , и выбираем из него вершину  $a$ . На шаге 3 полагаем резерв вершины  $b$  равным  $\min(7, \infty) = 7$  и устанавливаем вершину  $a$  в качестве предшественника вершины  $b$ . Помещаем вершину  $b$  во множество  $S$ . Устанавливаем резерв вершины  $d$  равным  $\min(6, \infty) = 6$  и устанавливаем вершину  $a$  в качестве предшественника вершины  $d$ . Помещаем вершину  $d$  во множество  $S$ . Шаги 4 и 5 не применяем, поэтому возвращаемся к шагу 2.

Проверяем, не является ли множество  $S$  пустым, и выбираем из него вершину, например,  $b$ . Теперь множество  $S$  содержит только вершину  $d$ . На шаге 3 полагаем резерв вершины  $c$  равным  $\min(3, 7) = 3$  и устанавливаем вершину  $b$  в качестве предшественника вершины  $c$ . Помещаем вершину  $c$  во множество  $S = \{c, d\}$ . Опять шаги 4 и 5 не применяем и возвращаемся к шагу 2.

Проверяем, не является ли множество  $S$  пустым, и выбираем из  $S$  вершину, например,  $c$ . Множество  $S$  опять содержит только вершину  $d$ . На шаге 3 полагаем резерв вершины  $z$  равным  $\min(3, 5) = 3$  и устанавливаем вершину  $c$  в качестве предшественника вершины  $z$ . Не помещаем  $z$  во множество  $S$ . На шаге 4 следует пометить вершину  $d$ , но она уже помечена.

На шаге 5 видим, что вершина  $z$  уже помечена, и, используя функцию предшествования, находим цепь

$$a \xrightarrow{(7,0)} b \xrightarrow{(3,0)} c \xrightarrow{(5,0)} z.$$

Добавляя 3 (резерв вершины  $z$ ) к потоку каждого ребра, получаем

$$a \xrightarrow{(7,3)} b \xrightarrow{(3,3)} c \xrightarrow{(5,3)} z,$$

что дает сеть, изображенную на рис. 2.10.

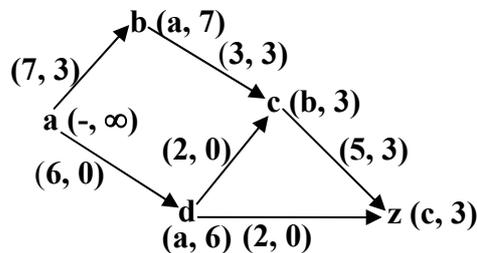


Рис. 2.10

Теперь возвращаемся к шагу 1, где переустанавливаем метки и предшественников и полагаем  $S = \{a\}$ . На шаге 2 выбираем вершину  $a$  и множества  $S$ . На шаге 3 устанавливаем резерв вершины  $b$  равным  $\min(\infty, 7 - 3) = 4$  и устанавливаем  $a$  в качестве предшественника вершины  $b$ . Помещаем вершину  $b$  во множество  $S$ . Полагаем также резерв вершины  $d$  равным 6 и устанавливаем вершину  $a$  в качестве предшественника вершины  $d$ . Помещаем вершину  $d$  во множество  $S$ . Шаги 4 и 5 не применяем, поэтому возвращаемся к шагу 2. Выбираем вершину  $b$  из множества  $S$ . Поскольку поток из  $b$  в  $c$  равен пропускной способности из  $b$  в  $c$ , то вершину  $c$  не помечаем. Выбираем вершину  $d$  из множества  $S$  и продолжаем процесс, после чего получаем цепь.

$$a \xrightarrow{(6,0)} d \xrightarrow{(2,0)} z$$

и добавляя 2 (резерв вершины  $z$ ) к потоку каждого ребра, получаем цепь

$$a \xrightarrow{(6,2)} d \xrightarrow{(2,2)} z.$$

Текущая ситуация изображена на рис. 2.11.

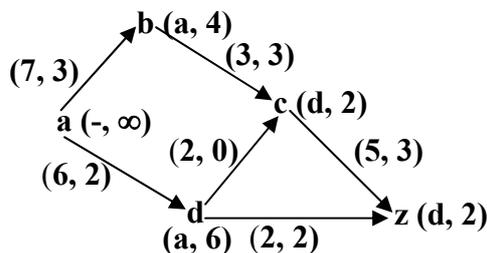


Рис. 2.11

Снова возвращаемся к шагу 1 и повторяем процесс, пока не получим цепь

$$a \xrightarrow{(6,2)} d \xrightarrow{(2,0)} c \xrightarrow{(5,3)} z.$$

Добавляя 2 (резерв вершины z) к потоку каждого ребра, получаем

$$a \xrightarrow{(6,4)} d \xrightarrow{(2,2)} c \xrightarrow{(5,5)} z.$$

Ситуация изображена на рис. 2.12.

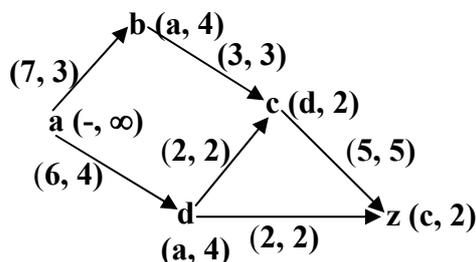


Рис. 2.12

Возвращаемся к шагу 1, переустанавливаем метки предшественника и полагаем  $S = \{a\}$ . На шаге 2 выбираем вершину  $a$  из множества  $S$ . Как и ранее, на шаге 3 полагаем резерв вершины  $b$  равным 4 и устанавливаем вершину  $a$  в качестве предшественника вершины  $b$ . Помещаем вершину  $b$  во множество  $S$ . Устанавливаем также резерв вершины  $d$  равным  $6 - 4 = 2$  и помещаем вершину  $d$  во множество  $S$ . Возвращаемся к шагу 2. Выбираем вершину  $b$  из множества  $S$ . Поскольку поток из вершины  $b$  в вершину  $c$  равен пропускной способности из  $b$  в  $c$ , пометить  $c$  нельзя, поэтому, в конце концов, возвращаемся к шагу 2. Выбираем вершину  $d$  из множества  $S$ . Поскольку поток из вершины  $d$  в вершину  $c$  и вершину  $z$  равен их пропускным способностям, нельзя пометить  $c$  и  $z$ . Опять возвращаемся к шагу 2, но множество  $S$  пусто. Процесс закончен. Окончательный поток изображен на рис. 2.13.

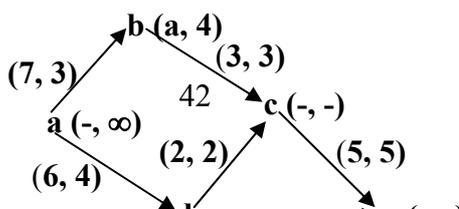


Рис. 2.13

**Пример.** Найдите максимальный поток для сети, изображенной на рис. 2.14.

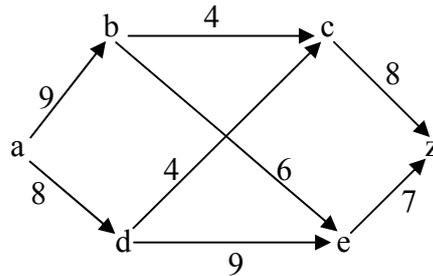


Рис. 2.14

Первые несколько проходов ничем не примечательны, поэтому опишем их вкратце. На первом проходе помечаем  $a(-, \infty)$ ,  $b(a, 9)$ ,  $d(a, 8)$ ,  $c(b, 4)$ ,  $e(b, 4)$  и  $z(c, 4)$ . Это дает  $a - z$  путь

$$a \xrightarrow{(9,0)} b \xrightarrow{(4,0)} c \xrightarrow{(8,0)} z,$$

и поскольку резерв z равен 4, добавляем 4 к каждому потоку, получая

$$a \xrightarrow{(9,4)} b \xrightarrow{(4,4)} c \xrightarrow{(8,4)} z,$$

что изображено на рис. 2.15

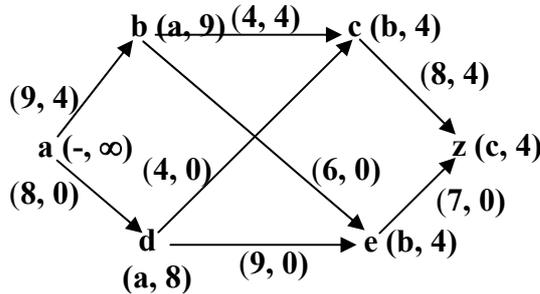


Рис. 2.15

На втором проходе помечаем  $a(-, \infty)$ ,  $b(a, 5)$ ,  $d(a, 8)$ ,  $c(d, 4)$ ,  $e(d, 8)$  и  $z(e, 7)$ . Это дает  $a - z$  путь

$$a \xrightarrow{(8,0)} d \xrightarrow{(9,0)} e \xrightarrow{(7,0)} z,$$

и поскольку резерв вершины  $z$  равен 7, добавляем 7 к каждому потоку, получая

$$a \xrightarrow{(8,7)} d \xrightarrow{(9,7)} e \xrightarrow{(7,7)} z,$$

что изображено на рис. 2.16

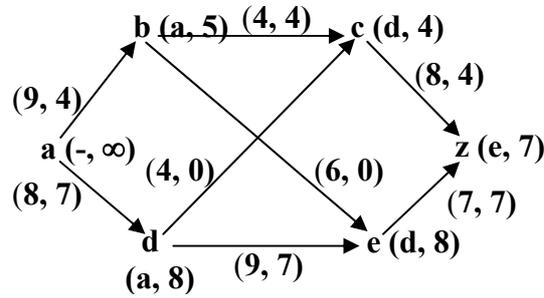


Рис. 2.16

На третьем проходе помечаем  $a(-, \infty)$ ,  $b(a, 5)$ ,  $d(a, 1)$ ,  $c(d, 1)$ ,  $e(d, 1)$  и  $z(c, 1)$ . Это дает  $a - z$  путь

$$a \xrightarrow{(8,7)} d \xrightarrow{(4,0)} c \xrightarrow{(8,4)} z,$$

и поскольку резерв  $z$  равен 1, добавляем 1 к каждому потоку, получая

$$a \xrightarrow{(8,8)} d \xrightarrow{(4,1)} c \xrightarrow{(8,5)} z,$$

что изображено на рис. 2.17.

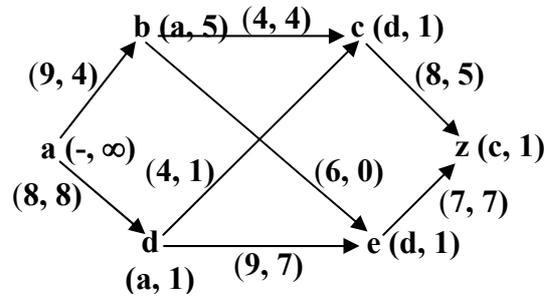


Рис. 2.17

Теперь начинается кое-что поинтересней, поэтому дальнейшее описание будет более подробным. Пометим  $b(a, 5)$ , но  $d$  пометить нельзя; помещаем  $b$  в  $S$ . При выборе вершины  $b$  из множества  $S$  нельзя пометить вершину  $c$ , но можно пометить  $e(b, 5)$ ; помещаем вершину  $c$  во множество  $S$ . Выбирая  $e$  из  $S$ , нельзя пометить вершину  $z$ , но т.к.  $d$  еще не помечена, ее можно пометить. Поскольку ребро  $(d, e)$  ориентировано неправильно, полагаем резерв вершины  $d$  равным  $\min(7, 5) = 5$ . Устанавливаем  $e$  в качестве предшественника вершины  $d$  и помещаем вершину  $d$  во множество  $S$ . Выбирая вершину  $d$  из множества  $S$ , имеем единственную возможность – пометить вершину  $c(d, 3)$  и поместить  $c$  в  $S$ . Выбираем вершину  $c$  из множества  $S$  и помечаем  $z(c, 3)$ . Это дает  $a - z$  путь

$$a \xrightarrow{(9,4)} b \xrightarrow{(6,0)} e \xrightarrow{(9,7)} d \xrightarrow{(4,1)} c \xrightarrow{(8,5)} z,$$

и поскольку резерв  $z$  равен 3, прибавляем 3 к каждому потоку, исключая ребро  $(d, e)$ . Это ребро ориентировано неправильно; вычитаем 3 из его потока, получая

$$a \xrightarrow{(9,7)} b \xrightarrow{(6,3)} e \xrightarrow{(9,4)} d \xrightarrow{(4,4)} c \xrightarrow{(8,8)} z,$$

что изображено на рис. 2.18.

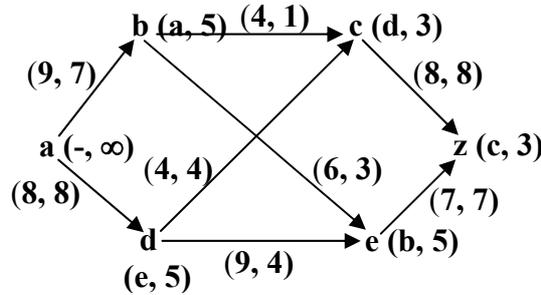


Рис. 2.18

Начинаем последний проход. Сначала  $S = \{a\}$ . Удаляем вершину  $a$  из множества  $S$ , помечаем вершину  $b(a, 2)$  и помещаем  $b$  в  $S$ . Удаляем вершину  $b$  из множества  $S$ , помечаем  $e(b, 2)$ ,  $c(b, 2)$  и помещаем вершины  $c, e$  во множество  $S$ . Удаляем  $e$  из  $S$ , помечаем  $d(e, 2)$ , поскольку резерв  $d$  равен минимуму из потока ребра  $(d, e)$  и резерва вершины  $d$ . Помещаем  $d$  в  $S$ . Затем из  $S$  удаляем  $d$ , но ни один из оставшихся шагов нельзя выполнить, поэтому возвращаемся к шагу 2. Следующим из  $S$  удаляем  $c$ , но мы не можем выполнить ни один из оставшихся шагов и возвращаемся к шагу 2. Множество  $S$  пусто, поэтому алгоритм завершен. Результат изображен на рис. 2.19

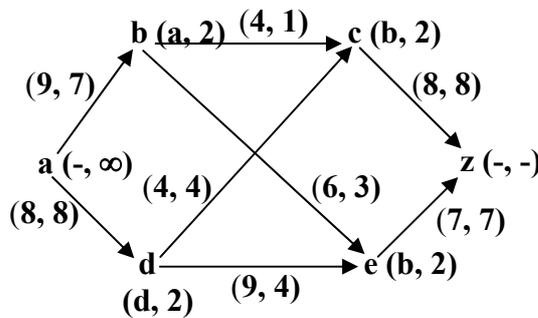


Рис. 2.19

**ТЕОРЕМА 2.3.** Алгоритм максимального потока строит максимальный поток для сети.

**ТЕОРЕМА 2.4.** Поток  $f$  на заданной сети  $N$  является максимальным тогда и только тогда, когда существует сечение  $(S, T)$ , такое, что  $val(f) = C(S, T)$ .

## 2.2 Сети Петри

В этом пункте продолжается рассмотрение двудольных графов с ориентированными ребрами. Однако, ориентация ребер допускается в обоих направлениях. Пусть  $G(V, E)$  – двудольный ориентированный граф, в котором  $V = P \cup T$ . Множество  $P$  называется множеством позиций, а множество  $T$  – множеством переходов.  $E$  – множество ориентированных ребер между  $P$  и  $T$ . В рассмотрение включено также множество функций  $M$  таких, что каждая функция  $\mu \in M$  ставит в соответствие неотрицательное целое число каждому элементу множества  $P$ . Функция  $\mu$  называется *разметкой* графа  $G$ . Ориентированный граф, обладающий такими свойствами, называется *сетью Петри*.

Сети Петри используются главным образом для моделирования параллельных процессов: для моделирования компонентов компьютера, параллельных вычислений, в робототехнике и даже для описания музыкальных структур. Вообще, сети Петри используются для нахождения дефектов в проекте системы, хотя имеют и многие другие применения. Они обладают многими свойствами блок-схем и конечных автоматов, речь о которых пойдет дальше. Более подробно.

Каждая позиция в сети Петри обозначена кружком, а каждый переход – вертикальной линией. Если  $p$  – позиция, то  $f(p)$  называется разметкой позиции  $p$ . Разметка множества  $G$  показана на графе с помощью больших черных точек, называемых метками, помещенных в кружки, которые обозначают позиции. Если кружок позиции  $p$  пуст, это означает, что в  $p$  меток нет, поэтому  $f(p) = 0$ .

Рассмотрим граф, изображенный на рис. 2.20. В этой сети Петри  $p_1$  и  $p_2$  – позиции, а  $t_1$  – переход. Позиция  $p_1$  содержит одну метку, а  $p_2$  – меток не содержит.

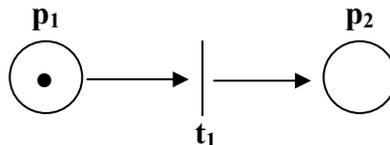


Рис. 2.20

*Срабатывание* перехода  $t$  – это удаление по одной метке из каждой позиции  $p_i$ , так что имеется ориентированное ребро из  $p_i$  в  $t$ , и добавление метки в каждую позицию  $p_i$ , так что имеется ориентированное ребро из  $t$  в  $p_i$ .

**Например**, срабатывание  $t_1$  в приведенном выше примере приводит к сети Петри, изображенной на рис. 2.21.

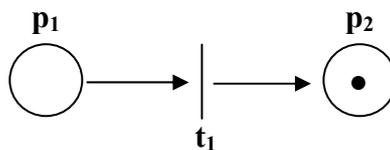


Рис. 2.21

В сети Петри, изображенной на рис. 2.22, срабатывание перехода  $t_1$  даст в результате сеть Петри, изображенную на рис. 2.23.

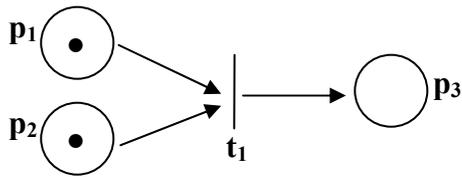


Рис. 2.22

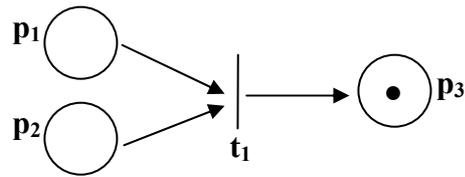


Рис. 2.23

В сети Петри, изображенной на рис. 2.24, срабатывание перехода  $t_1$  даст в результате сеть Петри, изображенную на рис. 2.25

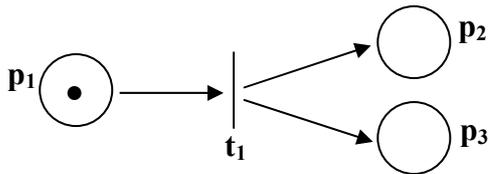


Рис. 2.24

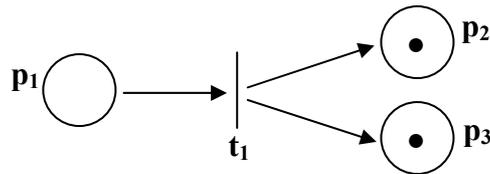


Рис. 2.25

В сети Петри, изображенной на рис. 2.26, срабатывание перехода  $t_1$  даст в результате сеть Петри, изображенную на рис. 2.27, поскольку имеется ориентированное ребро из  $t_1$  обратно в  $p_1$ .

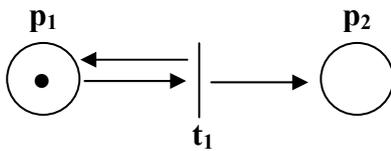


Рис. 2.26

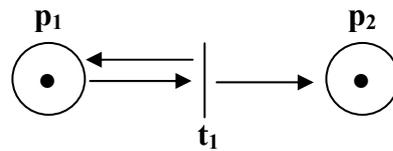


Рис. 2.27

Переход  $t$  может сработать, когда каждая позиция  $p_i$  такая, что имеется ориентированное ребро из  $p_i$  в  $t$ , содержит метку. Когда такая ситуация имеет место, говорят, что переход  $t$  разрешен. Переход  $t_1$  в сети Петри, изображенной на рис. 2.28, не может сработать, поскольку в  $p_2$  нет метки.

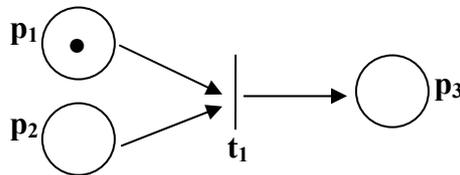


Рис. 2.28

В некоторых случаях необходимо, чтобы при срабатывании перехода удалялась или прибавлялась более чем одна метка. Если при срабатывании перехода  $t$  из позиции  $p$  необходимо удалить более чем одну метку, то ориентированное ребро из  $p$  в  $t$  помечается таким количеством меток, которое следует удалить из  $p$ . Если меток нет, то их количество по умолчанию предполагается

равным 1. Если при срабатывании перехода  $t$  в позицию  $p$  необходимо добавить более чем одну метку, то ориентированное ребро из  $t$  в  $p$  помечается таким количеством меток, которое нужно добавить в  $p$ . Если меток нет, то количество опять предполагается равным 1. Если при срабатывании перехода  $t$  из позиции  $p$  следует удалить  $n$  меток, то  $p$  должна содержать, по крайней мере,  $n$  меток, чтобы переход  $t$  был разрешен. В сети Петри, изображенную на рис. 2.29, срабатывание перехода  $t_1$  дает в результате сеть Петри, изображенную на рис. 2.30. Переход  $t_1$  не может сработать снова, поскольку в  $p_1$  нет достаточного количества меток.

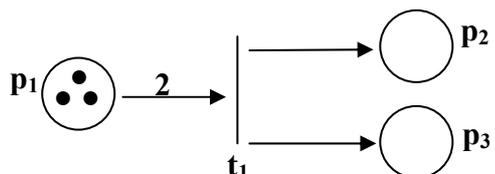


Рис. 2.29

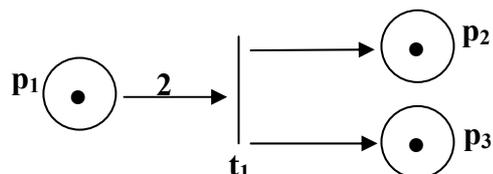


Рис.2.30

Срабатывание перехода предполагается мгновенным и может произойти в любой момент, когда переход разрешен. Порядок срабатывания переход ограничен только наличием достаточного количества меток, необходимых для разрешенности перехода. Вполне возможно, что в некоторый момент времени более чем один переход будет разрешен. В этом случае любой из них может сработать первым. Например, если моделируются несколько процессоров, имеющих общие данные и периферию, то неизвестно, когда одному из процессоров понадобится принтер или данные. Сети Петри используются для проверки работоспособности системы даже в условиях такой неопределенности системы.

**Состоянием**  $s$  сети Петри называется количество меток в каждой позиции, определяемое функцией  $\mu_i \in M$ . Таким образом, состояние сети Петри – это то же самое, что и разметка сети Петри. Термин «состояние» используется при рассмотрении автоматов и других машин с конечным количеством состояний. Поскольку  $\mu_i$  полностью определяют состояние, будем их отождествлять. Срабатывание перехода изменяет текущее состояние на состояние, определяемое другой функцией  $\mu_i$  из  $M$ . Поэтому новое состояние сети Петри зависит от текущего состояния и сработавшего перехода. Таким образом,  $\mu_i = \delta(\mu_i, t)$ , где  $\delta(\mu_i, t)$  – состояние, полученное из состояния  $\mu_i$ . Если в некоторый момент времени не существует разрешенного перехода, то функция состояния более не определяется, и процесс, описывающий сеть Петри, завершается.

Рассмотрим сеть Петри, изображенную на рис. 2.31. При срабатывании перехода  $t_1$  сеть Петри переходит в состояние, показанное на рис. 2.32. При срабатывании перехода  $t_4$  имеем сеть Петри, изображенную на рис. 2.33.

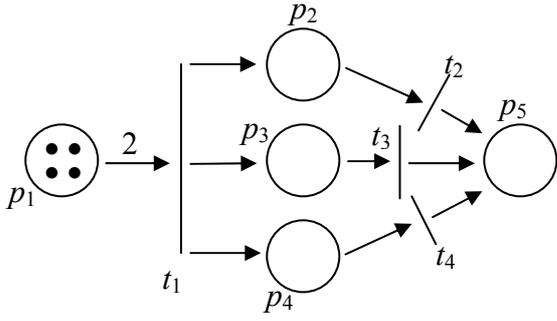


Рис. 2.31

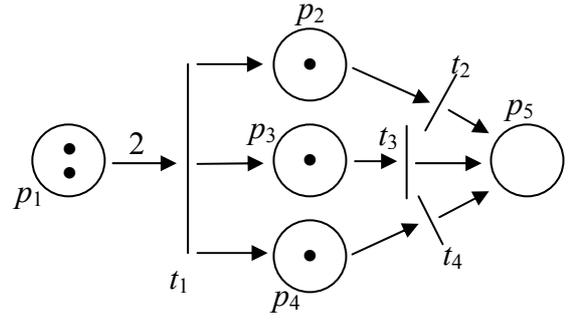


Рис.2.32

Вполне очевидно, что в конце концов все метки переместятся в  $p_5$ , и процесс завершится. Процесс в сети Петри, изображенной на рис. 2.34, никогда не завершится, поскольку переход  $t_1$  всегда разрешен.

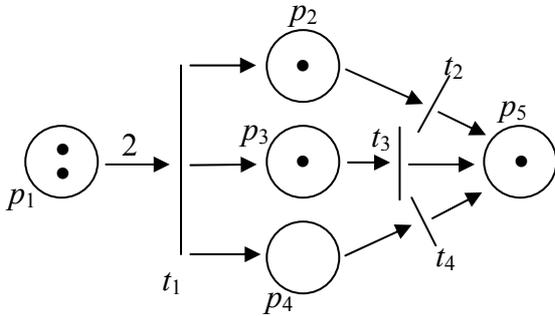


Рис. 2.33

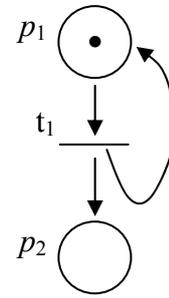


Рис.2.34

Сеть Петри на рис. 2.35 выполняет вычисления  $(a + b) \times (c + d)$ . Каждый из переходов  $t_1$  или  $t_2$  не сработают.

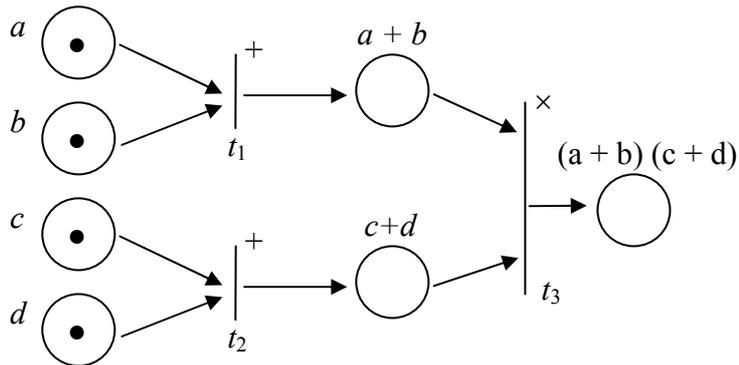


Рис. 2.35

Состояние  $\mu_i$  называется *непосредственно достижимым* из состояния  $\mu_i$ , если срабатывание какого-либо перехода  $t$  в то время, пока сеть находится в состоянии  $\mu_i$ , приводит к состоянию  $\mu_j$ . Более формально, состояние  $\mu_j$  непосредственно достижимо из состояния  $\mu_i$ , если существует переход  $t$  такой, что  $\mu_j =$

$\delta(\mu_i, t)$ . Состояние  $\mu_j$  *достижимо* из состояния  $\mu_i$ , если, начиная из состояния  $\mu_i$ , срабатывание последовательности переходов производит состояние  $\mu_j$ . Состояние  $\mu_{i_m}$  *достижимо* из состояния  $\mu_{i_1}$ , если существуют состояния  $\mu_{i_2}, \mu_{i_3}, \dots, \mu_{i_{m-1}}$ , и переходы  $t_1, t_2, \dots, t_{m-1}$  такие, что  $\mu_{i_{k+1}} = \delta(\mu_{i_k}, t_k)$  для  $k = 1, 2, \dots, m - 1$ . В частности, состояние, изображенное на рис. 2.36, непосредственно *достижимо* из состояния, изображенного на рис. 2.37, поскольку оно производится срабатыванием перехода  $t_1$ .

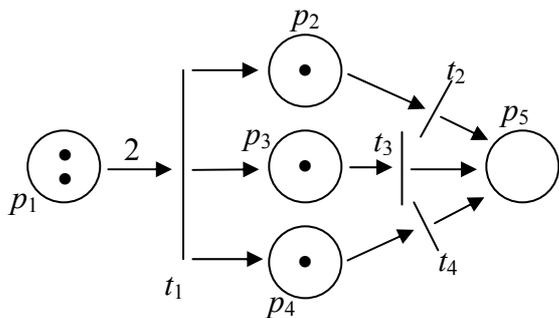


Рис. 2.36

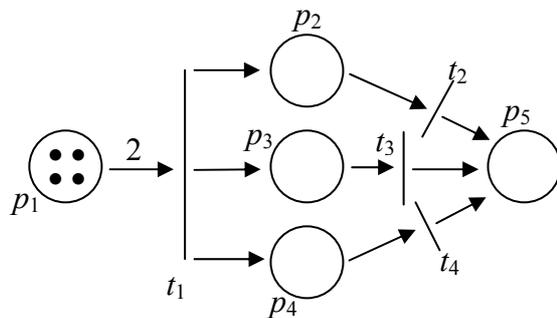


Рис. 2.37

Состояние сети Петри, изображенное на рис. 2.38, *достижимо* из состояния сети Петри, изображенного на рис. 2.39, поскольку оно производится срабатыванием перехода  $t_1$ , а затем срабатыванием перехода  $t_4$ .

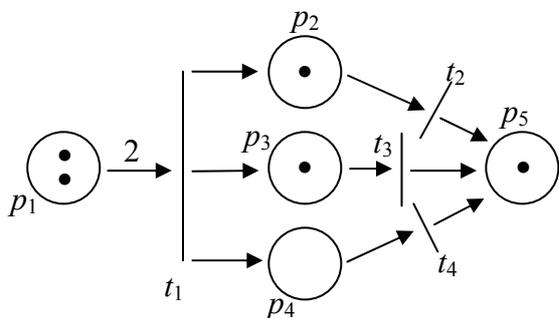


Рис. 2.38

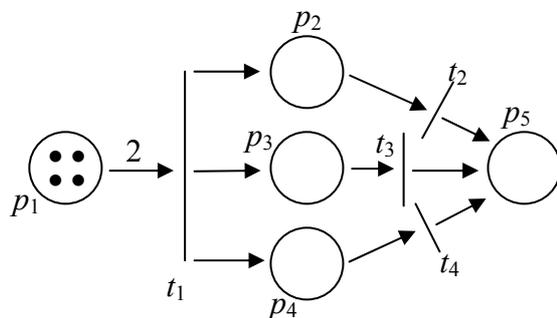


Рис. 2.39

Сеть Петри называется *живой*, если для любого состояния  $\mu$  и любого перехода  $t$  существует состояние  $\mu'$ , *достижимое* из состояния  $\mu$ , так что переход  $t$  в состоянии  $\mu'$  является разрешенным. Таким образом, каким бы ни было текущее состояние, существует такая последовательность срабатывания переходов с началом в текущем состоянии, что любой заданный переход может сработать. Очевидно, что сеть на рис. 2.40 является *живой*.

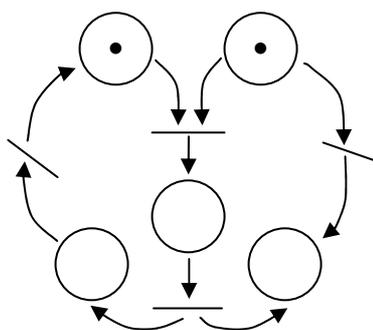


Рис. 2.40

Одно из определений тупиковой сети Петри – сеть не живая. Следовательно, существует такое состояние, что если сеть Петри находится в этом состоянии, то один или более переходов могут никогда не сработать. Назовем это состояние *частичным тупиком*. Будем говорить, что сеть Петри *находится в тупике*, если имеется состояние, в котором ни один из переходов не может сработать. Таким образом, существует такое состояние  $\mu$ , когда  $\delta(\mu, t)$  не определена для всех переходов  $t$ .

**Пример.** Рассмотрим ситуацию, когда два процессора используют совместно два ресурса, например, принтер и память, каждый из которых не может быть использован ими одновременно. Если один процессор имеет доступ к принтеру, а другой – к памяти, то оба не могут завершить задания. Они также не могут освободить используемый ресурс, поэтому система находится в тупике. Эта ситуация показана на рис. 2.41.

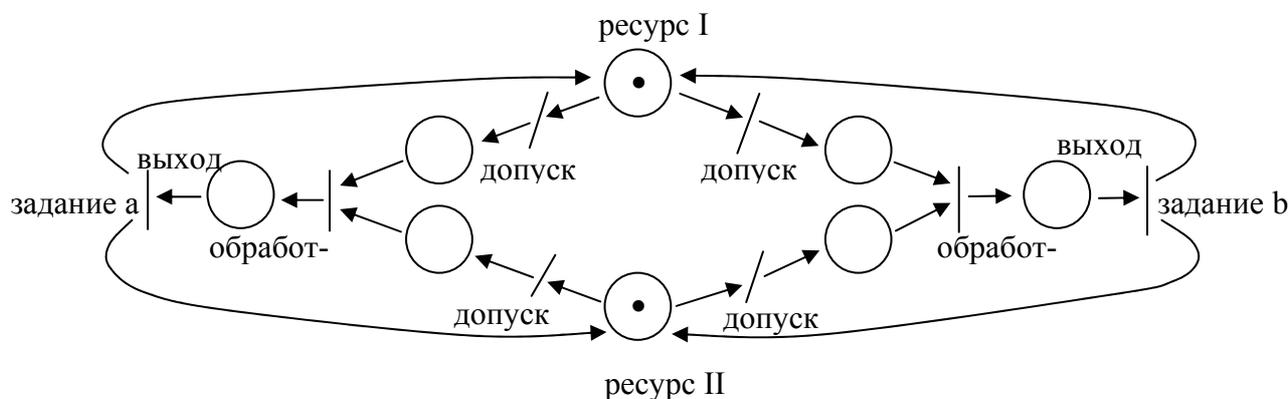


Рис. 2.41

Еще один пример, *проблему обедающих философов*, мы рассмотрим в упражнениях.

Другая, с которой приходится сталкиваться при использовании сети Петри для совместного использования файлов, – это проблема взаимного исключения. Предположим, что одновременный доступ двух людей к одним и тем же данным нежелателен. Так, недопустимо, чтобы один человек читал данные, а другой в это же время их изменял. Эта проблема решается путем взаимного исключения. Пока один человек работает с данными, для другого доступ закрыт. Такая ситуация проиллюстрирована на рис. 2.42. Единственная метка в позиции

с препятствует допуску другого субъекта до тех пор, пока метка не будет перемещена.

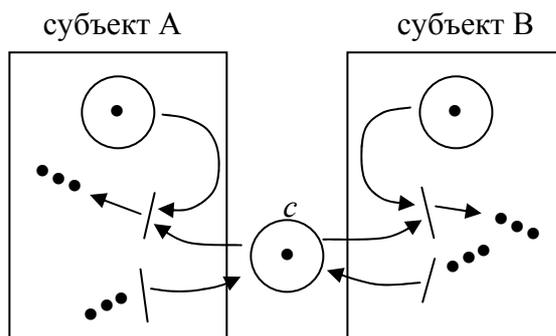


Рис. 2.42

Сеть Петри называется *безопасной*, если каждая позиция содержит не более одной метки. Когда сеть Петри безопасна, в каждой позиции имеется либо одна метка, либо они отсутствуют вообще, поэтому помещение метки в позицию является бинарной операцией. Для большинства моделей контроля необходима безопасная сеть Петри, поскольку наличие метки может означать протекание процесса, а ее отсутствие – сигнал о его остановке.

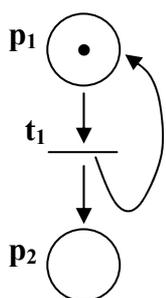


Рис. 2.43

Сеть Петри называется *ограниченной*, если количество меток в каждой позиции не превышает некоторое целое число  $k$ . Ограниченная сеть Петри предоставляет возможность контроля проблемы переполнения. Очевидно, что безопасная сеть Петри – ограниченная. Сеть Петри, изображенная на рис. 2.43, ограниченной не является.

В некоторых сетях Петри на вместимость позиций накладываются ограничения, и по условию количество меток не может превышать эту вместимость.

Сеть Петри называется *консервативной*, если общее число меток во всех позициях всегда постоянно. В этом случае количество меток на входе каждого перехода равно количеству меток на выходе перехода. Если метки описывают ресурсы, то консервативная сеть Петри гарантирует, что никакой ресурс не будет ни создан, ни утерян. Консервативная сеть Петри является очевидным образом ограниченной. Сеть Петри, изображенная на рис. 2.44, консервативна.

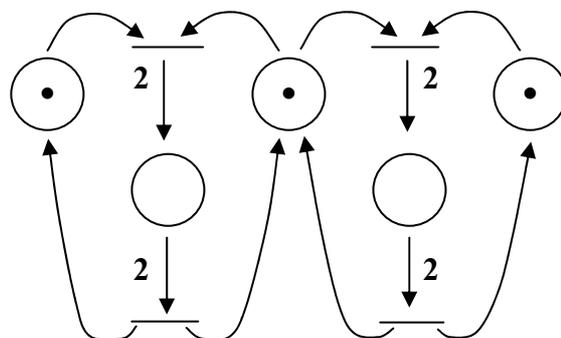


Рис. 2.44

## УПРАЖНЕНИЯ

1. Для сети, изображенной на рис. 2.45,
- а) проверить сохранение потока в вершинах  $b$ ,  $c$  и  $d$ ;
  - б) найти  $val(f)$ , величину потока;
  - в) найти значение  $C(S, T)$ , где  $S = \{a, b, c, d\}$ ;
  - г) найти значение  $C(S, T)$ , где  $S = \{a, b, e\}$ ;
  - д) найти значение  $C(S, T)$ , где  $S = \{a, b, d\}$ ;

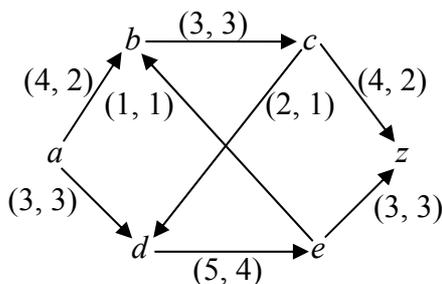


Рис. 2.45

2. Для сети, изображенной на рис. 2.46,
- а) проверить сохранение потока в вершинах  $b$ ,  $d$  и  $e$ ;
  - б) найти  $val(f)$ , значение потока;
  - в) найти значение  $C(S, T)$ , где  $S = \{a, b, c, d\}$ ;
  - г) найти значение  $C(S, T)$ , где  $S = \{a, b, d, e\}$ ;
  - д) найти значение  $C(S, T)$ , где  $S = \{a, b, d\}$ ;

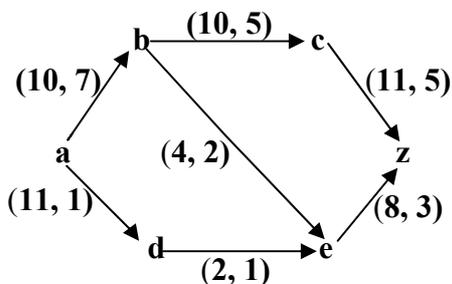


Рис. 2.46

3. Дополнить поток в сети, изображенной на рис. 2.47, так чтобы имело место сохранение потока.

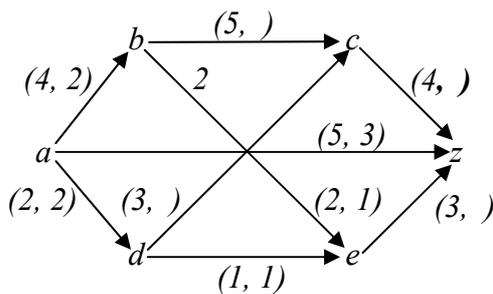


Рис. 2.47

4. Дополнить поток в сети, изображенной на рис. 2.48, так чтобы имело место сохранение потока.

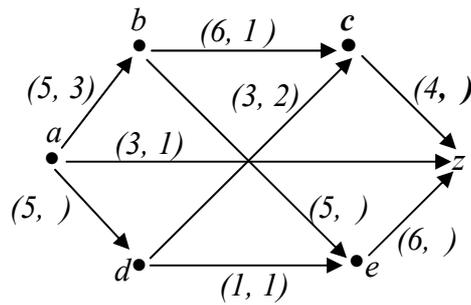


Рис. 2.48

5. Найти максимальный поток в сети, изображенной на рис. 2.49.

6. Найти максимальный поток в сети, изображенной на рис. 2.50.

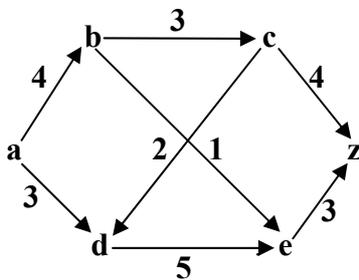


Рис. 2.49

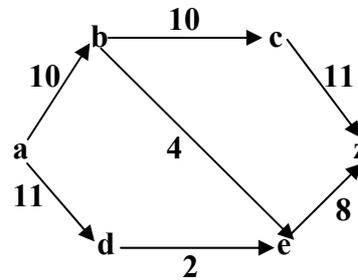


Рис. 2.50

7. Найти максимальный поток в сети, изображенной на рис. 2.51

8. Найти максимальный поток в сети, изображенной на рис. 2.52.

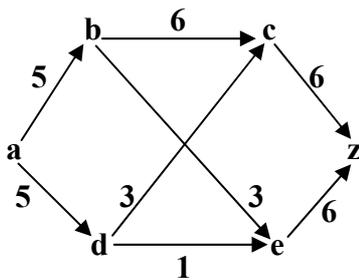


Рис. 2.51

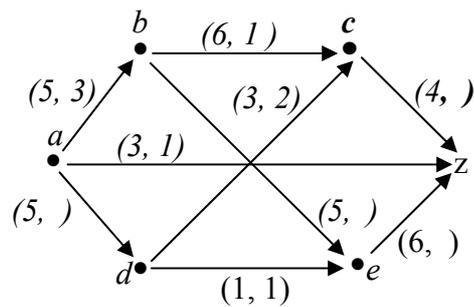


Рис. 2.52

9. Найти минимальное мечение для потока в сети из упражнения 5.

10. Найти минимальное сечение для потока в сети и упражнения 6.

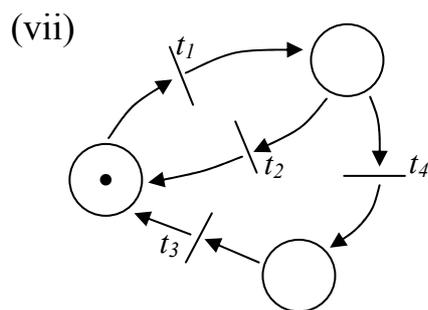
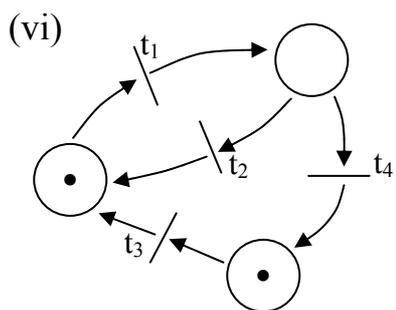
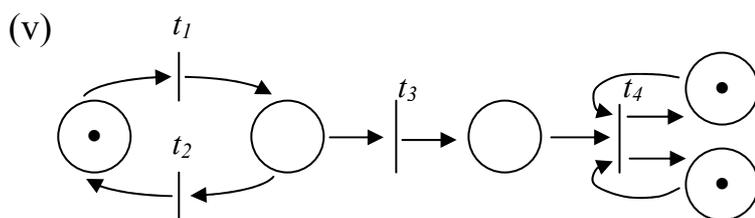
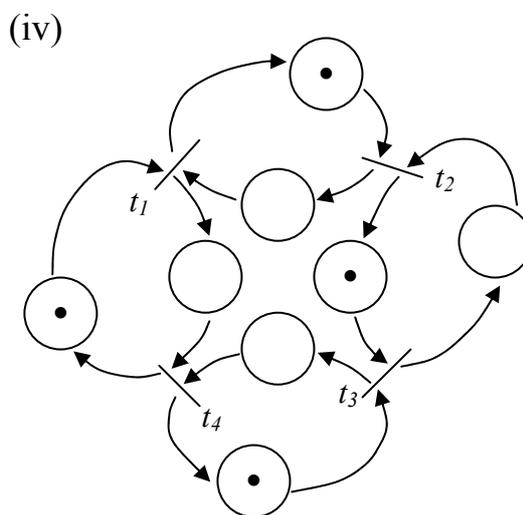
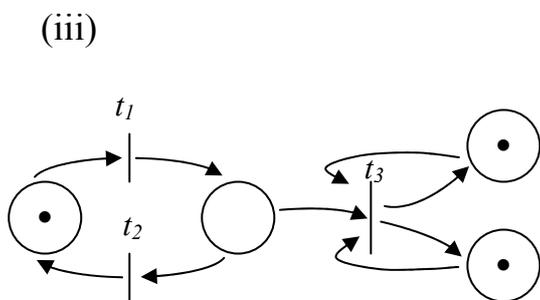
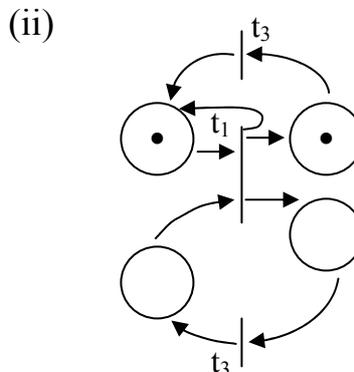
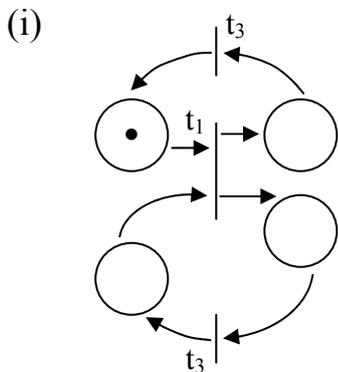
11. Найти минимальное сечение – разрез для потока в сети из упражнения 7.

12. Найти минимальное сечение – разрез для потока в сети из упражнения 8.

13. Какие из приведенных ниже сетей Петри

- являются безопасными?
- являются консервативными?
- являются ограниченными?

- г) имеют достижимыми каждое состояние?
- д) являются живыми?
- е) имеют все текущие переходы разрешенными?
- ж) имеют переходы, которые могут быть разрешены в некотором состоянии, достижимом из текущего состояния?



14. В сетях Петри (упражнение 13) найдите:
  - а) все возможные состояния сети Петри;
  - б) состояние после срабатывания перехода  $t_1$ ;
  - в) состояние после последовательного срабатывания переходов  $t_1$  и  $t_2$ ;
  - г) любое тупиковое состояние;
  - д) любое частично тупиковое состояние.
15. Постройте пример сети Петри, в которой два перехода разрешены, но срабатывание каждого из них блокирует другой.
16. Постройте пример сети Петри, которая является безопасной, но не является живой.
17. Постройте пример сети Петри с ограничением 4.
18. Постройте пример сети Петри, которая является консервативной, но не является ограниченной.
19. Постройте пример сети Петри, которая является консервативной, но не является безопасной.
20. Постройте сеть Петри, которая эквивалентна сети, изображенной на рис. 2.41, но является тупиковой.
21. Может ли консервативная сеть Петри быть тупиковой?
22. Приведите пример сети Петри, которая имеет состояние, недостижимое из исходного состояния.
23. Постройте сеть Петри, которая частично тупиковая, но не тупиковая.
24. Проблема обедающих философов формулируется следующим образом: пять философов сидят за столом, чтобы есть и размышлять. В любой заданный промежуток времени они могут делать что-то одно. Философы едят китайское блюдо, используя палочки, поэтому для еды каждому необходимо иметь две палочки. Когда философы размышляют, палочки не нужны. На столе между двумя сидящими лежит одна палочка, т.е. всего пять палочек. Очевидно, что если философы одновременно возьмут правую палочку, то каждый будет лишен пищи. Постройте сеть Петри, моделирующую эту задачу, которая не была бы тупиковой, т.е. когда философ будет приниматься за еду, у него для этого будут две палочки по обе стороны от него.

### 3. ОСНОВНЫЕ ТИПЫ ЗАДАЧ И СПОСОБЫ ИХ РЕШЕНИЯ

- *Сетевая модель* – графическое изображение плана выполнения комплекса работ, состоящего из нитей (работ) и узлов (событий), которые отражают логическую взаимосвязь всех операций. В основе сетевого моделирования лежит изображение планируемого комплекса работ в виде графа. Граф-схема, состоящая из заданных точек (вершин), соединенных системой линий. Отрезки, соединяющие вершины, называются ребрами (дугами) графа. Ориентированным называется такой граф, на котором стрелкой указаны направления всех его ребер (дуг), что позволяет определить, какая из двух его граничных вершин является начальной, а какая – конечной. Исследование таких сетей проводится методами теории графов.

Теория графов оперирует понятием пути, объединяющим последовательность взаимосвязанных ребер. Контур означает такой путь, у которого начальная вершина совпадает с конечной. Сетевой график – это ориентированный граф без контуров. В сетевом моделировании имеются два основных элемента работа и событие.

- *Работа* – это активный процесс, требующий затрат ресурсов, либо пассивный (ожидание), приводящий к достижению намеченного результата.

- *Фиктивная работа* – это связь между результатами работ (событиями), не требующая затрат времени и ресурсов.

- *Событие* – это результат (промежуточный или конечный) выполнения одной или нескольких предшествующих работ.

- *Путь* – это любая непрерывная последовательность (цепь) работ и событий.

- *Критический путь* – это путь, не имеющий резервов и включающий самые напряженные работы комплекса. Работы, расположенные на критическом пути, называют критическими. Все остальные работы являются не критическими (ненапряженными) и обладают резервами времени, которые позволяют передвигать сроки их выполнения, не влияя на общую продолжительность выполнения всего комплекса работ.

При построении сетевых моделей необходимо соблюдать следующие правила.

1. Сеть изображается слева направо, и каждое событие с большим порядковым номером изображается правее предыдущего. Общее направление стрелок, изображающих работы, также в основном должно быть расположено слева направо, при этом каждая работа должна выходить из события с меньшим но-

мером и входить в событие с большим номером.

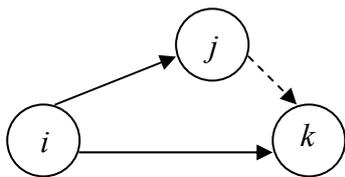


Рис. 3.1

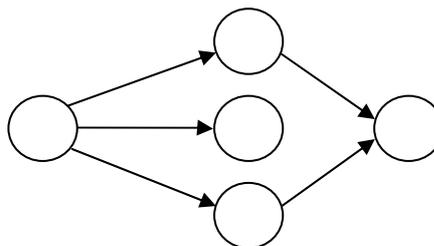


Рис. 3.2

2. Два соседних события могут объединяться лишь одной работой. Для изображения параллельных работ вводятся промежуточное событие и фиктивная работа (рис. 3.1).

3. В сети не должно быть тупиков, т.е. промежуточных событий, из которых не выходит ни одна работа (рис. 3.2).

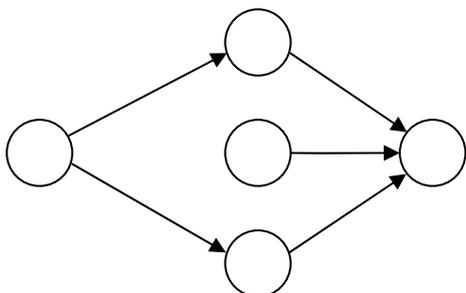


Рис. 3.3

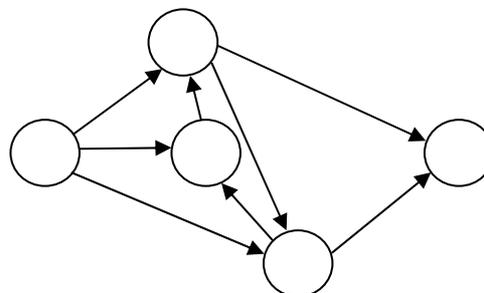


Рис. 3.4

4. В сети не должно быть промежуточных событий, которым не предшествует хотя бы одна работа (рис. 3.3).

5. В сети не должно быть замкнутых контуров, состоящих из взаимосвязанных работ, создающих замкнутую цепь (рис. 3.4). Для правильной нумерации событий поступают следующим образом: нумерация событий начинается с исходного события, которому дается номер 1. Из исходного события 1 вычеркивают все исходящие из него работы, на оставшейся сети вновь находят событие, в которое не входит ни одна работа. Этому событию дается номер 2. Затем вычеркивают работы, выходящие из события 2, и вновь находят на оставшейся части сети событие, в которое не входит ни одна работа, ему присваивается номер 3, и так продолжается до завершающего события. Пример нумерации сетевого графика показан на рис. 3.5.

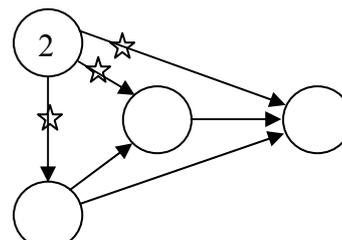
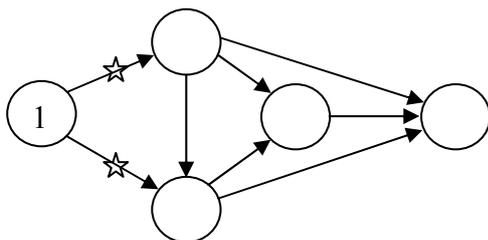




Рис. 3.5

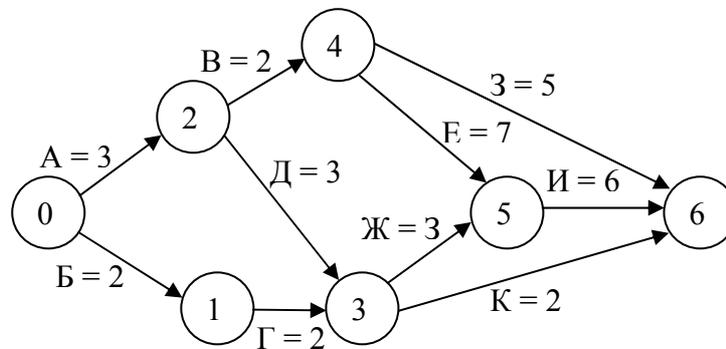


Рис. 3.6

Продолжительность выполнения работ устанавливается на основании действующих нормативов или по экспертным оценкам специалистов. В первом случае временные оценки являются детерминированными (однозначными), во втором – стохастическими (вероятностными).

Рассмотрим в качестве примера программу создания нового бытового прибора, пользующегося спросом у населения. Необходимые данные приведены в табл. 3.1.

На основании данных таблицы построим сетевой график создания прибора с учетом выше изложенных рекомендаций (рис. 3.6).

### 3.1 Расчет временных параметров сетевого графика

Основным временным параметром сетевого графика является продолжительность критического пути.

Расчет критического пути включает два этапа. Первый называется прямым проходом. Вычисления начинают с исходного события и продолжают до тех пор, пока не будет достигнуто завершающее событие. Для каждого события определяется одно число, представляющее ранний срок его наступления. На втором этапе, называемом обратным проходом, вычисления начинают с завершающего события и продолжают, пока не будет достигнуто исходное событие. Для каждого события вычисляется поздний срок его наступления.

Рассмотрим прямой проход:

Таблица 3.1

Операции	Наименование работы	Непосредственно предшествующие операции	Продолжительность, недели
А, Б	Разработка технической документации (ТД) на прибор и его электронную часть.	—	А – 3, Б - 2
В, Г	Разработка технологической документации на электронную часть прибора и прибор.	А, Б	В – 2, Г - 2
Д	Передача ТД на прибор.	А	3
Е	Изготовление приборов	В	7
Ж	Изготовление электронной части прибора	Д, Г	3
З, К	Разработка ТД на эксплуатацию прибора и электронную часть.	Г, В	3 – 5, К - 2
И	Сборка и испытания прибора	Е, Ж	6

$t_i^{p.H}$  – ранний срок начала всех операций, выходящих из событий  $i$ .

Если  $i = 0$ , то  $t_0^{p.H} = 0$ ;

$t_j^{p.H}$  – ранний срок начала всех операций, входящих в  $j$ .

Тогда

$$t_j^{p.H} = \max_i (t_i^{p.H} + t_{ij}) \text{ для всех } (i, j),$$

где  $t_{ij}$  – продолжительность операции  $(i, j)$ ;

$$t_1^{p.H} = t_0^{p.H} + t_{0,1} = 0 + 2 = 2; \quad t_2^{p.H} = t_0^{p.H} + t_{0,2} = 0 + 3 = 3;$$

$$t_3^{p.H} = \max_{i=1,2} \{2 + 2, 3 + 3\} = 6; \quad t_4^{p.H} = \max_{i=2,3} \{3 + 2, 6 + 0\} = 6;$$

$$t_5^{p.H} = \max_{i=3,4} \{6 + 3, 6 + 7\} = 13; \quad t_6^{p.H} = \max_{i=3,4,5} \{6 + 2, 6 + 5, 13 + 6\} = 19;$$

Прямой проход закончился, начинаем обратный:

$t_i^{n.o}$  – поздний срок окончания всех операций, входящих в событие  $i$ .

Если  $i = n$ , где  $n$  – завершающее событие сети, то  $t_n^{n.o} = t_n^{p.H}$  и является отправной точкой обратного прохода;

$$t_i^{n.o} = \min_j (t_j^{n.o} - t_{i,j}) \text{ для всех операций } (i, j);$$

$$t_6^{n.o} = t_6^{p.H} = 19;$$

$$t_5^{n.o} = t_6^{n.o} - t_{5,6} = 19 - 6 = 13;$$

$$t_4^{n.o} = \min_{j=5,6} \{19 - 5, 13 - 7\} = 6;$$

$$t_3^{n.o} = \min_{j=4,5,6} \{19 - 2, 13 - 3, 6 - 0\} = 6;$$

$$t_2^{n.o} = \min_{j=3,4} \{6 - 3, 6 - 2\} = 3;$$

$$t_1^{n.o} = t_3^{n.o} - t_{1,3} = 6 - 2 = 4;$$

$$t_0^{n.o} = \min_{j=1,2} \{3 - 3, 4 - 2\} = 0.$$

Используя результаты вычислений при прямом и обратном проходах, можно определить операции критического пути. Операция  $(i, j)$  принадлежит критическому пути, если она удовлетворяет условиям:

$$t_i^{p.H} = t_i^{n.o},$$

$$t_j^{p.H} = t_j^{n.o},$$

$$t_j^{p.H} - t_i^{p.H} = t_j^{n.o} - t_i^{n.o} = t_{ij}.$$

Для рассматриваемого примера критический путь включает операции (0, 2), (2, 3), (3, 4), (4, 5), (5, 6).

Операции связаны еще с двумя сроками:

$t_{ij}^{n.H}$  – поздний срок начала работы. Он является наиболее поздним (максимальным) из допустимых моментов начала данной работы, при котором еще возможно выполнение всех последующих работ в установленный срок:

$$t_{ij}^{n.H} = t_j^{n.o} - t_{ij};$$

$t_{ij}^{p.o}$  – ранний срок окончания работы. Он является наиболее ранним (минимальным) из возможных моментов окончания работы при заданной продолжительности работ:

$$t_{ij}^{p.o} = t_i^{p.H} + t_{ij}.$$

Различают два вида резервов времени: полный резерв ( $r_n$ ) и свободный резерв ( $r_{св}$ ).

*Полный резерв времени* показывает, на сколько может быть увеличена сумма продолжительности всех работ относительно критического пути. Он представляет собой разность между максимальным отрезком времени, в течение которого может быть выполнена операция, и ее продолжительностью ( $t_{ij}$ ) и определяется как

$$t_{ij}^{n.H} - t_i^{p.H}.$$

*Свободный резерв времени* – максимальное время, на которое можно отсрочить начало или увеличить продолжительность работы при условии, что все события наступают в ранние сроки:

$$r_{свij} = t_j^{p.H} - t_i^{p.H} - t_{ij}.$$

Результаты расчета критического пути и резервов времени некритических операций представлены в нижеследующей таблице. Следует отметить, что критические операции должны иметь нулевой полный резерв времени, при этом свободный резерв также должен быть равен нулю.

### 3.2 Построение сетевого графика и распределение ресурсов

Конечным результатом выполняемых на сетевой модели расчетов является сетевой график (план). При построении сетевого графика необходимо учитывать наличие ресурсов, так как одновременное выполнение некоторых операций из-за ограничений, связанных с рабочей силой, оборудованием и другими видами ресурсов, иногда оказывается невозможным. Именно в этом отношении представляют ценность полные резервы времени некритических операций.

Сдвигая некритическую операцию в том или ином направлении, но в пределах ее полного резерва времени, можно добиться снижения максимальной потребности в ресурсах. Однако даже при отсутствии ограничений на ресурсы полные резервы времени обычно используются для выравнивания потребностей в ресурсах на протяжении всего срока реализации программы работ. Это означает, что работы удастся выполнить более или менее постоянным составом рабочей силы.

Таблица 3.2

Работа (i,j)	Продолжительность $t_{ij}$	Раннее		Позднее		Полный резерв $r_n$	Свободный резерв $r_{св}$
(0, 1)	2	0	2	2	4	2	0
(0, 2)	3	0	3	0	3	0 <sup>к</sup>	0
(1, 3)	2	2	4	4	6	2	2
(2, 3)	3	3	6	3	6	0 <sup>к</sup>	0
(2, 4)	2	3	5	4	6	1	1
(3, 4)	0	6	6	6	6	0 <sup>к</sup>	0
(3, 5)	3	6	9	10	13	4	4
(3, 6)	2	6	8	17	19	11	11
(4, 5)	7	6	13	6	13	0 <sup>к</sup>	0
(4, 6)	5	6	11	14	19	8	8
(5, 6)	6	13	19	13	19	0 <sup>к</sup>	0

**Примечание:** к – критические операции.

На рис. 3.7 показан график примера, рассмотренного выше. Роль полных и свободных резервов при выборе сроков объясняется двумя правилами:

1) если полный резерв равен свободному, то календарные сроки не критической операции можно выбрать в любой точке между ее ранним началом и поздним окончанием;

2) если свободный резерв меньше полного, то срок начала не критической операции можно сдвинуть по отношению к раннему сроку ее начала не более чем на величину свободного резерва.

В вышеуказанном примере правило 2 применимо к операции (0, 1), а сроки всех остальных операций выбираются по правилу 1.

В табл. 3.2 представлены ресурсы рабочей силы для различных операций.

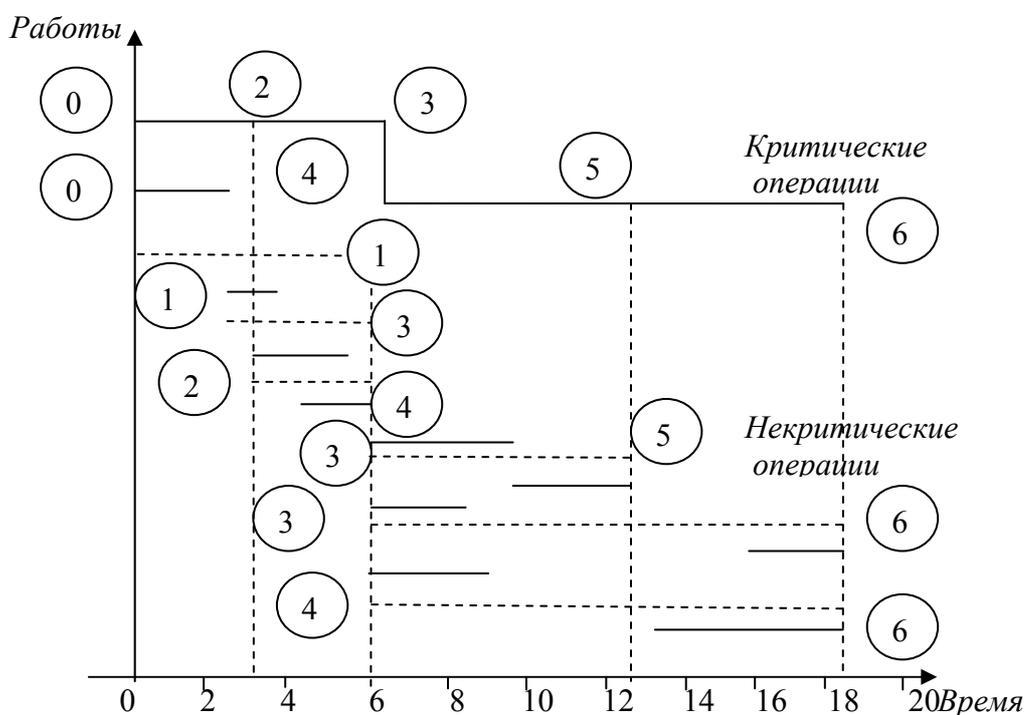


Рис. 3.7

На рис. 3.8 показана потребность в рабочей силе при условии выбора в качестве календарных сроков не критических операций начала их ранних сроков, на рис. 3.9 – потребность в рабочей силе при выборе наиболее поздних сроков.

Пунктирной линией представлена потребность критических операций, которая должна быть удовлетворена, если нужно выполнить все работы в минимально возможный срок.

Оптимальное решение задачи равномерного использования ресурсов (минимизация максимальной потребности в ресурсах) представлено на рис. 3.10, уточненный график выполнения работ указан на рис. 3.11.

### 3.3 Учет стоимостных факторов при реализации сетевого графика

Стоимостные факторы при реализации сетевого графика учитываются путем определения зависимости «затраты – продолжительность» для каждой операции. При этом рассматриваются прямые затраты, а косвенные типа административных или управленческих расходов не принимаются во внимание.

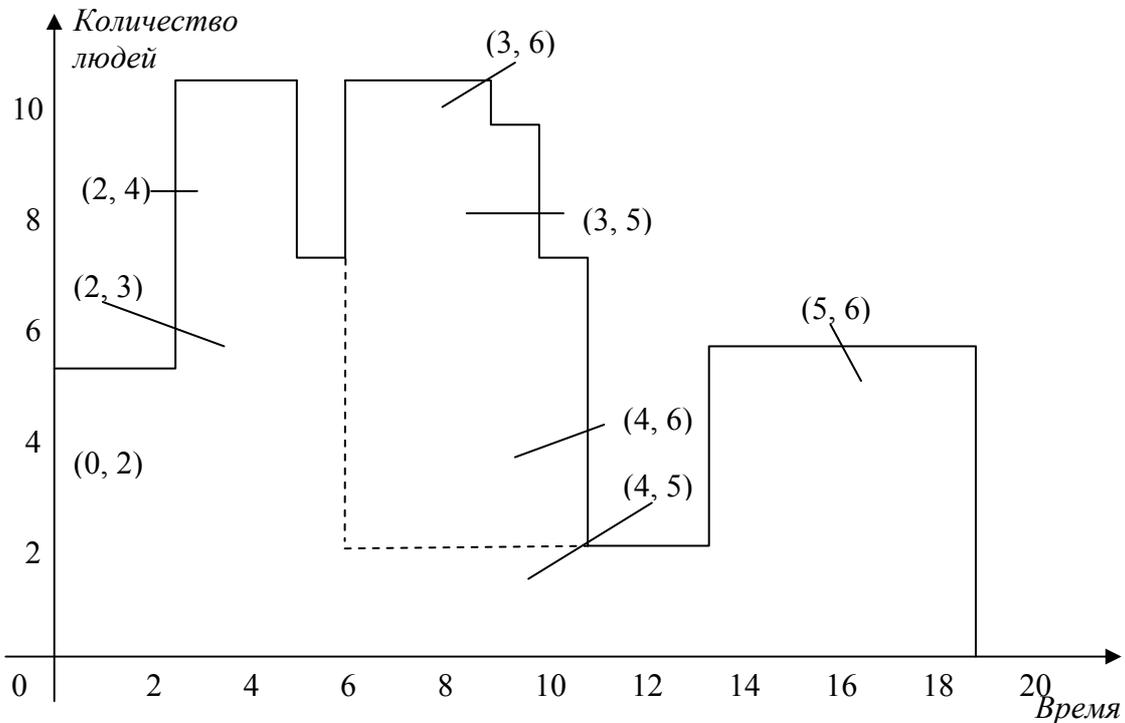


Рис. 3.8

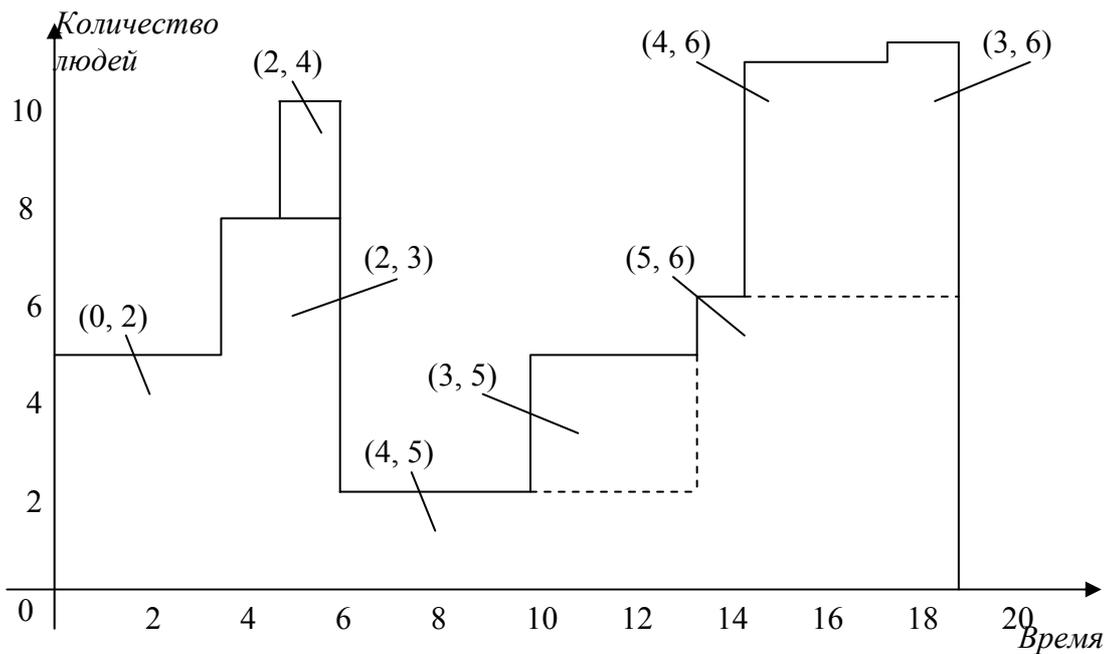


Рис. 3.9

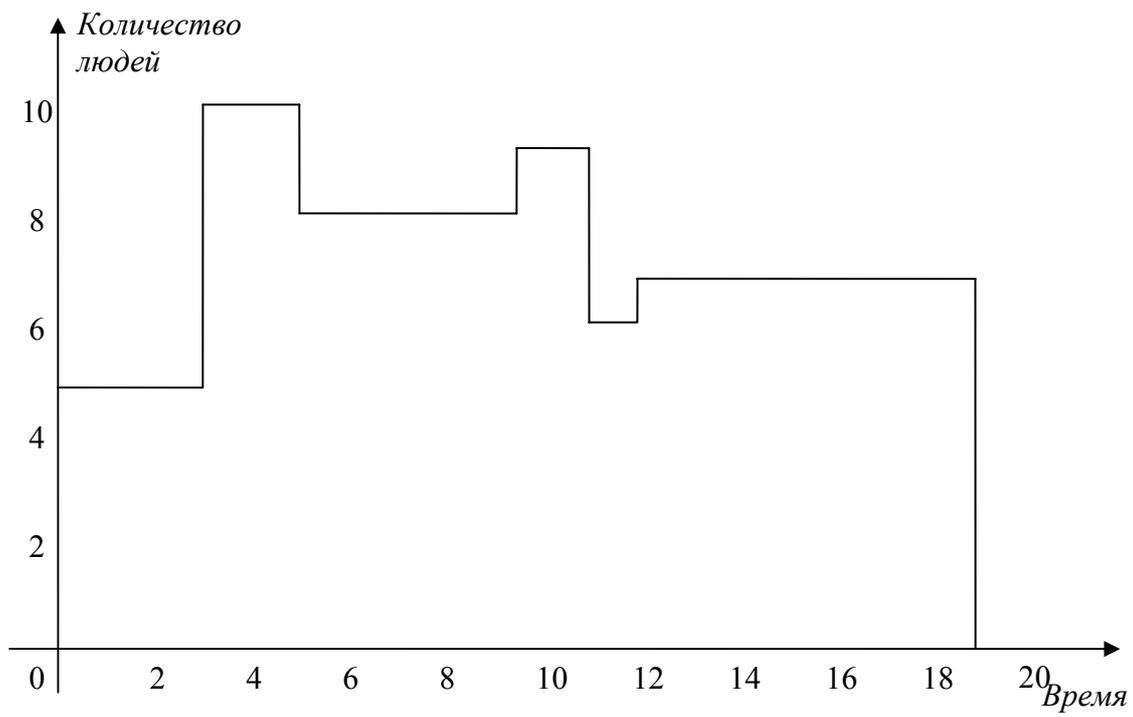


Рис. 3.10

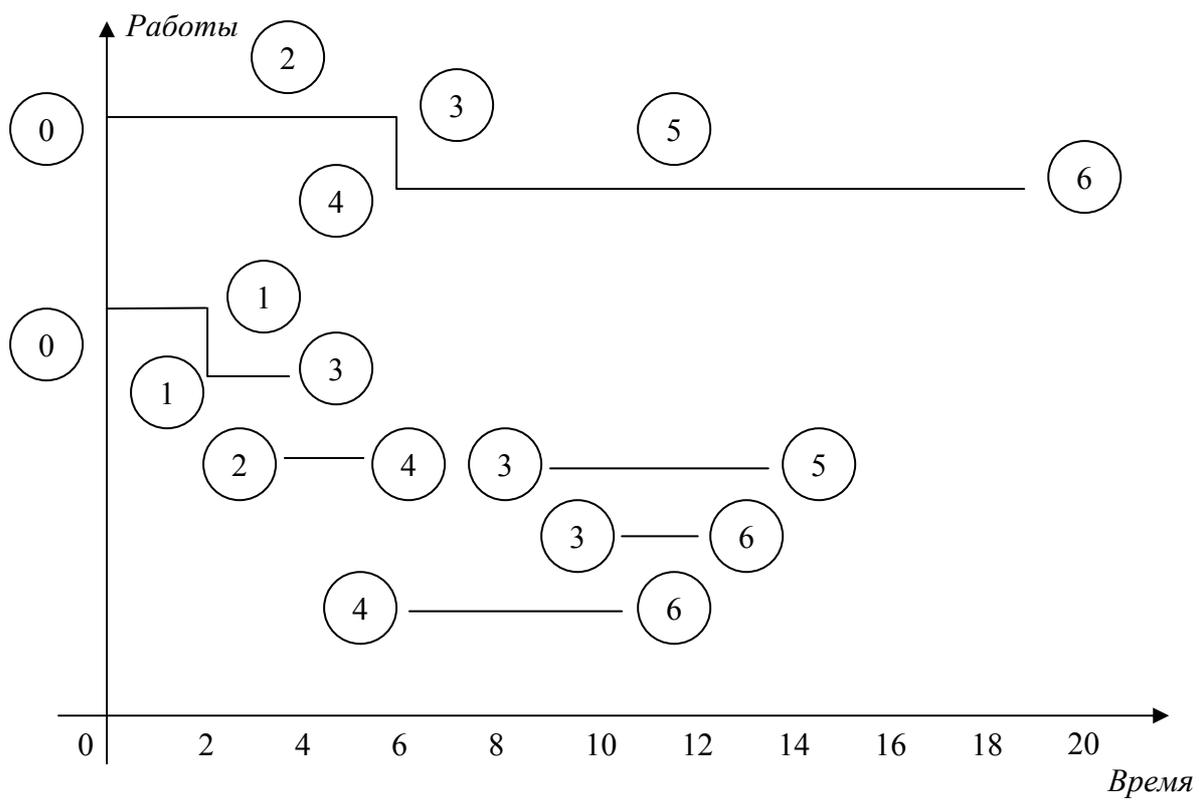


Рис. 3.11



Рис. 3.12

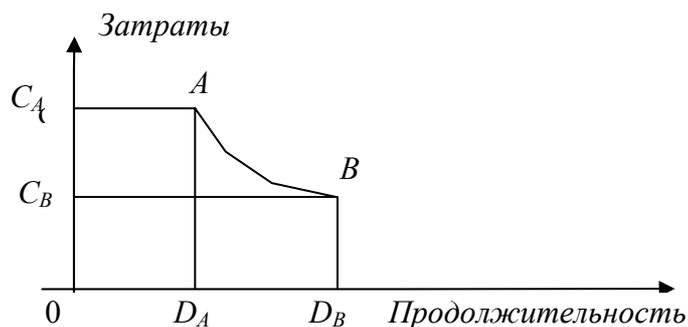


Рис. 3.13

На рис. 3.12 показана линейная зависимость стоимости операции от ее продолжительности. Точка  $(D_B, C_B)$ , где  $D_B$  — продолжительность операции, а  $C_B$  — ее стоимость, соответствует нормальному режиму выполнения операции. Продолжительность операции можно уменьшить (сжать), увеличив интенсивность использования ресурсов, а следовательно, увеличив стоимость операции. Однако существует предел, называемый *минимальной продолжительностью операции*. За точкой, соответствующей этому пределу (точка максимально интенсивного режима), дальнейшее увеличение интенсивности использования ресурсов ведет лишь к увеличению затрат без сокращения продолжительности операции. Этот предел обозначен на рис. 3.12 точкой  $A$  с координатами  $(D_A, C_A)$ .

Линейная зависимость «затраты — продолжительность» принимается из соображений удобства, так как ее можно определить для любой операции по двум точкам нормального и максимально интенсивного режимов, т.е. по точкам  $A$  и  $B$ .

Использование нелинейной зависимости «затраты — продолжительность» существенно усложняет вычисления. Поэтому иногда нелинейную зависимость можно аппроксимировать кусочно-линейной (рис. 3.13), когда операция разбивается на части, каждая из которых соответствует одному линейному отрезку. Следует отметить, что наклоны этих отрезков при переходе от точки нормального режима к точке максимально интенсивного режима возрастают. Если это условие не выполняется, то аппроксимация не имеет смысла.

Определив зависимость «затраты — продолжительность», для всех операций сети принимают нормальную продолжительность. Далее рассчитывается

сумма затрат на все операции сети при этой продолжительности работ. На следующем этапе рассматривается возможность сокращения продолжительности работ. Этого можно достичь за счет уменьшения продолжительности какой-либо критической операции, только критические операции и следует подвергать анализу.

Чтобы добиться сокращения продолжительности выполнения работ при минимально возможных затратах, необходимо в максимально допустимой степени сжать ту критическую операцию, у которой наклон кривой «затраты – продолжительность» наименьший. В результате сжатия критической операции получают новый календарный график, возможно, с новым критическим путем. Стоимость работ при новом календарном графике будет выше стоимости работ по предшествующему графику. На следующем этапе этот новый график вновь подвергается сжатию за счет следующей критической операции с минимальным наклоном кривой «затраты – продолжительность» при условии, что продолжительность этой операции не достигла минимального значения. Подобная процедура повторяется, пока все критические операции не будут находиться в режиме максимальной интенсивности. Полученный таким образом оптимальный календарный график соответствует минимуму прямых затрат.

### 3.4 Обоснование привлекательности проекта по выпуску продукции

Для финансирования проектов по строительству и наладке изготовления конкурентоспособной продукции в большинстве случаев фирмам требуются инвестиции. Включение в проект материалов с оптимизацией сетевых моделей в части обоснования сроков возврата инвестиций делает проект более привлекательным и способствует принятию инвестором положительного решения.

**Пример.** Предприятие решило для улучшения финансового состояния наладить выпуск конкурентоспособной продукции (мороженого). Для переоборудования цеха (участка) под выпуск этой продукции необходимо выполнить:

- 1) подготовку технического задания на переоборудование участка (30 дн.);
- 2) заказ и поставку нового оборудования (60 дн.);
- 3) заказ и поставку нового электрооборудования (50 дн.);
- 4) демонтаж старого и установку нового оборудования (90 дн.);
- 5) демонтаж старого и установку нового электрооборудования (80 дн.);
- 6) переобучение персонала (30 дн.);
- 7) испытания и сдачу в эксплуатацию оборудования для производства мороженого (20 дн.).

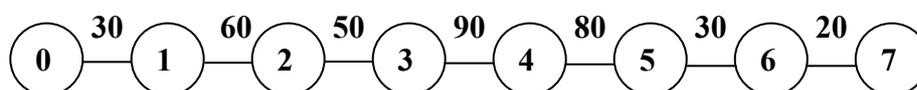
Ожидается, что производительность после ввода новой линии составит 20 т мороженого в смену. Прибыль от реализации 1 т продукции составит 0,5 тыс. р. в смену. Деньги на покупку и переоборудование участка в размере 2000 тыс. р. взяты в банке под 20% годовых (из расчета 1500 тыс. р. на закупку оборудования и 500 тыс. р. на работы по демонтажу старого оборудования и установке нового оборудования). Затраты на проведение работ в нормальном и максимальном режимах указаны в табл. 3.3.

Определить, через какое время может быть возвращен кредит в банк.

Таблица 3.3

Работа	Нормальный режим		Максимальный режим	
	Продолжительность, дн.	Затраты, тыс.р	Продолжительность, дн.	Затраты, тыс. р.
1	30	20	25	30
2	60	40	45	60
3	50	30	40	40
4	90	70	70	100
5	80	60	65	70
6	30	25	20	25
7	20	20	17	25
<b>Итого</b>	360	265	282	350

**РЕШЕНИЕ.** 1. Составим график проведения работ по пуску новой линии:



На проведение переоборудования необходимо

$$30 + 60 + 50 + 90 + 80 + 30 + 20 = 360 \text{ дн.}$$

2. График можно улучшить, выполняя некоторые работы параллельно. Получим график (рис. 3. 14).

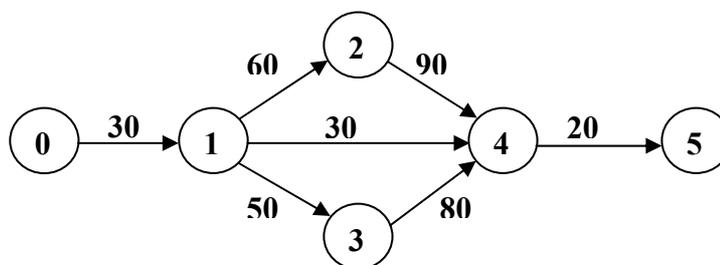


Рис. 3.14

На этом графике обозначены работы:

- 0, 1 – подготовка технического задания;
- 1, 2 – заказ и поставка нового оборудования;
- 1, 3 – заказ и поставка нового электрооборудования;
- 2, 4 – установка нового оборудования;
- 3, 4 – установка нового электрооборудования;
- 1, 4 – переобучение персонала;
- 4, 5 – сдача в эксплуатацию новой линии.

По графику путь (0, 1), (1, 2), (2, 4), (4, 5) имеет продолжительность 200 дн.; (0, 1), (1, 3), (3, 4), (4, 5) – 180 дн.; (0, 1), (1, 4), (4, 5) – 80 дн.

Критическим путем графика является путь, на котором находятся работы (0, 1), (1, 2),

(2, 4), (4, 5) продолжительностью

$$30 + 60 + 90 + 20 = 200 \text{ дн.}$$

График улучшился на  $360 - 200 = 160$  дн.

Определим, через какое время после начала выпуска мороженого может быть возвращен кредит в банк.

Через 200 дн. после начала работ предприятие истратит 1500 тыс. р. на приобретение оборудования (согласно условию примера) и 265 тыс. р. на его установку и сдачу в эксплуатацию (см. табл. 3.3, столбец «Затраты» при нормальном режиме). в наличии у предприятия останется

$$2000 - 1500 - 265 = 235 \text{ тыс. р.}$$

Построим графики изменения кредита в зависимости от времени получения прибыли предприятием – от выпуска мороженого (рис. 3.15).

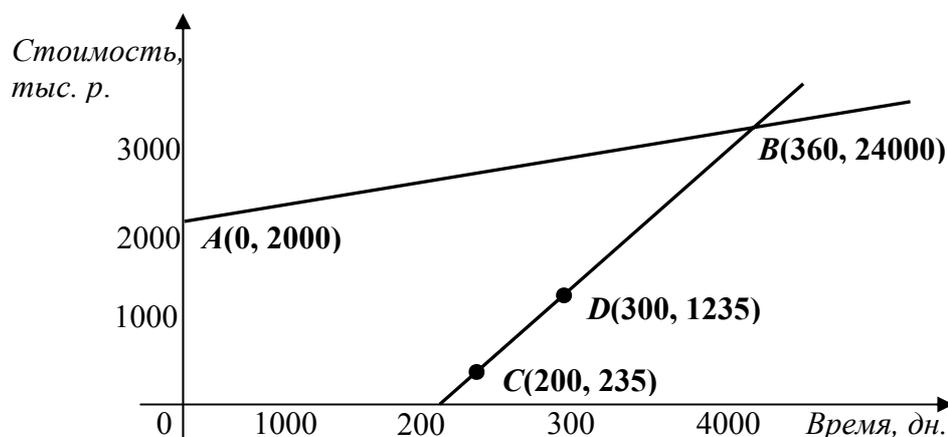


Рис. 3.15

Для построения графика изменения кредита в зависимости от времени составим уравнение. Через 360 дн. после выдачи банком кредита под 20% годовых долг предприятия составит 2400 тыс. р. Поэтому известны две точки этой прямой:  $A(0, 2000)$ ,  $B(360, 2400)$ . Согласно уравнению прямой, проходящей через две точки:

$$\frac{(y - y_A) / (y_B - y_A) = (x - x_A) / (x_B - x_A),}{(y - 2000) / (2400 - 2000) = (x - 0) / (360 - 0)}.$$

Решая уравнение, получим

$$10x - 9y + 18000 = 0 \tag{3.1}$$

Найдем уравнение прибыли предприятия. Известно, что через 200 дн. после начала работ у предприятия осталось от кредита 235 тыс. р. Через 100 дн. после начала выпуска продукции предприятие получит прибыль

$$0,5(\text{тыс. р.}) \cdot 20(\text{т}) \cdot 100(\text{дн.}) = 1000 \text{ тыс. р.}$$

и у него будет в наличии

$$1000 + 235 = 1235 \text{ тыс. р.}$$

Таким образом, для нахождения уравнения прибыли имеем две точки:  $C(200, 235)$ ,  $D(300, 1235)$ . Тогда

$$\begin{aligned} (y - y_C) / (y_D - y_C) &= (x - x_C) / (x_D - x_C), \\ (y - 235) / (1235 - 235) &= (x - 200) / (300 - 200), \\ 10x - y - 1765 &= 0. \end{aligned} \quad (3.2)$$

Решая совместно уравнения (3.1) и (3.2), определим время, когда кредит может быть возвращен в банк:

$$\begin{cases} 10x - 9y + 18000 = 0, \\ 10x - y - 1765 = 0. \end{cases}$$

Откуда получаем  $y = 2471$ ,  $x = 423,6 \approx 424$  дн.

3. График выполнения работ может быть сжат за счет выполнения некоторых операций в максимально интенсивном режиме.

Вычислим наклоны кривой «затраты – продолжительность» для каждой операции. Результаты расчетов даны в табл. 3.4.

Учитывая наклоны кривой, производим сжатие операций  $(0, 1)$ ,  $(2, 4)$ ,  $(3, 4)$ ,  $(4, 5)$ , получим – сетевой график (рис. 3.16).

Таблица 3.4

Операции	Наклон
0, 1	2
1, 2	1, 3
1, 3	1
2, 4	1, 5
3, 4	0, 7
1, 4	1
4, 5	1, 7

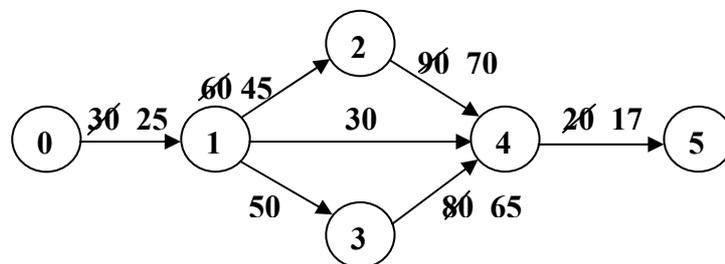


Рис. 3.16

Новый график имеет 2 критических пути:  $(0, 1)$ ,  $(1, 2)$ ,  $(2, 4)$ ,  $(4, 5)$  и  $(0, 1)$ ,  $(1, 3)$ ,  $(3, 4)$ ,  $(4, 5)$  с продолжительностью 157 дн.

Таким образом, критический путь сокращен с 200 до 157 дн., а это означает, что пред-

приятие начнет производить мороженое через 157 дн. после начала работ.

Определим, сколько предприятию придется заплатить за «сжатие» критического пути (см. табл. 3.3):

$$\begin{aligned}(0, 1) &: 30 - 20 = 10 \text{ тыс. р.}; \\(1, 2) &: 60 - 40 = 20 \text{ тыс. р.}; \\(2, 4) &: 100 - 70 = 30 \text{ тыс. р.}; \\(3, 4) &: 70 - 60 = 10 \text{ тыс. р.}; \\(4, 5) &: 25 - 20 = 5 \text{ тыс. р.}\end{aligned}$$

Таким образом, «сжатие» работ (0, 1), (1, 2), (2, 4), (3, 4), (4, 5) обойдется предприятию в

$$10 + 20 + 30 + 10 + 5 = 75 \text{ тыс. р.}$$

График изменения кредита в зависимости от времени остается прежним (см. рис. 3.15). Его вид определяет уравнение

$$10x - 9y + 18000 = 0.$$

Найдем уравнение прибыли.

Через 157 дн. после начала работ у предприятия осталось от кредита

$$2000 - 1500 - 265 - 75 = 160 \text{ тыс. р.}$$

Через 100 дн. после начала выпуска продукции предприятие получит прибыль

$$20(\text{тыс. р.}) \cdot 0.5(\text{тыс. р.}) \cdot 100(\text{дн.}) = 1000 \text{ тыс. р.},$$

и у него будет в наличии

$$1000 + 160 = 1160 \text{ тыс. р.}$$

Таким образом, для нахождения уравнения прибыли предприятия имеем две точки:

$$C'(157, 160) \text{ и } D'(257, 1160).$$

Согласно уравнению прямой, проходящей через 2 точки, получим

$$\begin{aligned}\frac{(y - y_C)}{(y_D - y_C)} &= \frac{(x - x_C)}{(x_D - x_C)}, \\ \frac{(y - 160)}{(1160 - 160)} &= \frac{(x - 157)}{(257 - 157)}, \\ 10x - y - 1410 &= 0.\end{aligned}\tag{3.3}$$

Решая совместно уравнения (3.1) и (3.3), определим время, когда кредит может быть возвращен в банк:

$$\begin{cases} 10x - 9y + 18000 = 0, \\ 10x - y - 1410 = 0, \end{cases} \\ y = 2426, 25, x = 383, 6 \approx 384 \text{ дн.}$$

Таким образом, через 384 дн. предприятие может вернуть кредит в банк. По сравнению с предыдущим случаем (см. п. 2) предприятие вернет в банк деньги раньше на  $424 - 384 = 40$  дн.

При нормальном режиме работ критический путь составляет 200 дн., стоимость работ = 265 тыс. р.

Критический путь уменьшен до 157 дн., минимальная стоимость работ составляет  $265 + 75 = 340$  тыс. р. при максимальном режиме.

### 3.5 Минимизация сети

Задача минимизации сети состоит в нахождении ребер, соединяющих все узлы сети и имеющих минимальную суммарную длину (рис. 3.17).

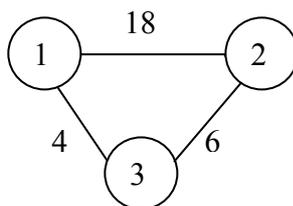


Рис. 3.17

На ребрах, соединяющих узлы 1, 2, 3, указаны длины. Узел 3 соединен с узлами 1 и 2 минимальной длиной  $4 + 6 = 10$ . Если соединить узлы 1 и 2, то возникает цикл и получающаяся сеть не будет минимальной. Отсутствие циклов в минимальной сети дало ей название «минимальное дерево – остов».

#### Алгоритм решения

Начнем с любого узла и соединим его с ближайшим узлом сети. Соединенные два узла образуют связное множество, а остальные – несвязное. Далее в несвязном множестве выберем узел, который расположен ближе других к любому из узлов связного множества. Скорректируем связное и несвязное множества и будем повторять процесс до тех пор, пока в связное множество не попадут все узлы сети. В случае одинаково удаленных узлов выберем любой из них, что указывает на неоднозначность (альтернативность) «минимального дерева – остова».

**Пример.** Телевизионная фирма планирует создание кабельной сети для обслуживания 5 районов-новостроек. Числа на ребрах указывают длину кабеля (рис. 3.18). Узел 1 – телевизионный центр. Отсутствие ребра между двумя узлами означает, что соединение соответствующих новостроек либо связано с большими затратами, либо невозможно.

Найти такое соединение кабелем районов-новостроек, чтобы длина его была минимальной.

**РЕШЕНИЕ.** Минимальная длина кабеля:  $1 + 3 + 4 + 3 + 5 = 16$  (рис. 2.19).

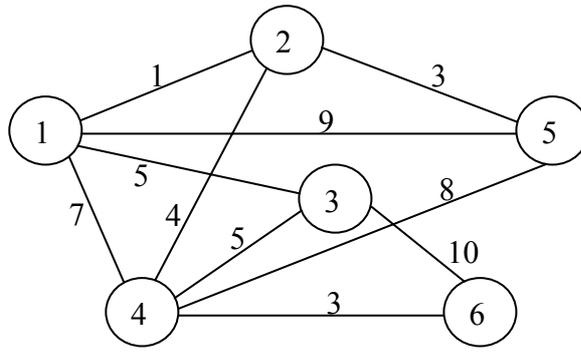


Рис. 3.18

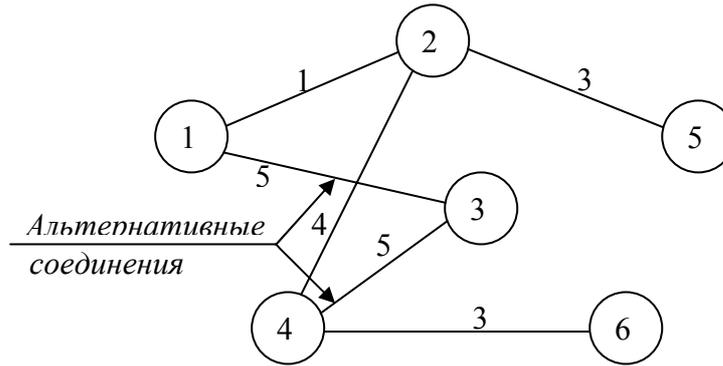


Рис. 3.19

**Пример.** На рис. 3.20 указаны длины коммуникаций, связывающих 9 установок по добыче газа в открытом море с расположенным на берегу приемным пунктом. Поскольку скважина 1 расположена ближе всех к берегу, она оснащена необходимым оборудованием для перекачки газа, идущего с остальных скважин в приемный пункт.

Построить сеть трубопровода, соединяющего все скважины с приемным пунктом и имеющего минимальную общую длину труб.

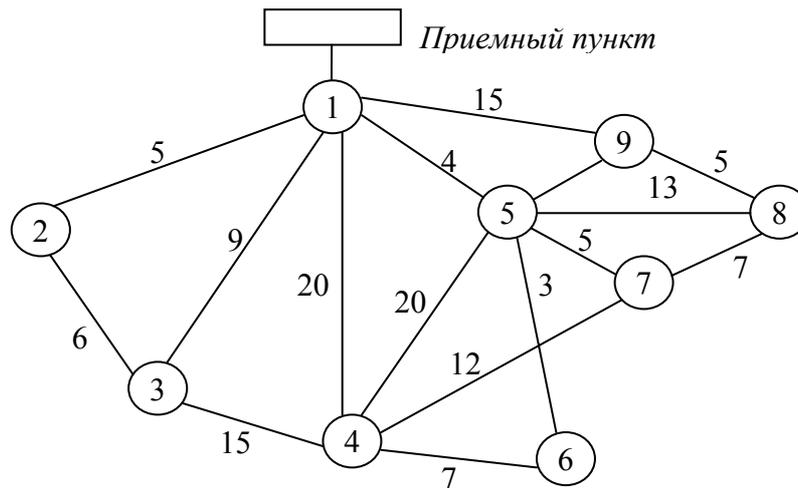


Рис. 3.20

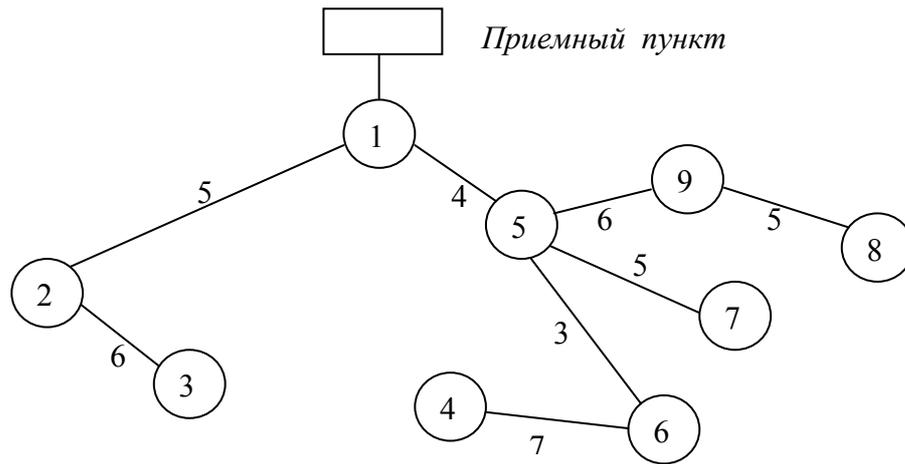


Рис. 3.21

**РЕШЕНИЕ.** Минимальная длина труб:  $5 + 6 + 4 + 3 + 7 + 5 + 6 + 5 = 41$  (рис. 3.21).

### *Нахождение кратчайшего пути*

Задача состоит в нахождении связанных между собой дорог на транспортной сети, которые в совокупности имеют минимальную длину от исходного пункта до пункта назначения.

Введем обозначения:

$d_{ij}$  – расстояние на сети между смежными узлами  $i$  и  $j$ ;

$U_j$  – кратчайшее расстояние между узлами  $i$  и  $j$ ,  $U_1 = 0$ .

Формула для вычисления  $U_j$ :

$$U_j = \min_i \left\{ \begin{array}{l} \text{Кратчайшее расстояние до предыду-} \\ \text{щего узла } i \text{ плюс расстояние} \\ \text{между текущим узлом } j \text{ и предыду-} \\ \text{щим узлом } i \end{array} \right\} = \min_i \{U_i + d_{ij}\}$$

Из формулы следует, что кратчайшее расстояние  $U_j$  до узла  $j$  можно вычислить лишь после того, как определено кратчайшее расстояние до каждого предыдущего узла  $i$ , соединенного дугой с узлом  $j$ . Процедура завершается, когда получено  $U_i$  последнего звена.

Определить кратчайшее расстояние между узлами 1 и 7 (рис. 3.22).

**РЕШЕНИЕ.** Найдем минимальные расстояния:

$$U_1 = 0,$$

$$U_2 = U_1 + d_{12} = 0 + 2 = 2,$$

$$U_3 = U_1 + d_{13} = 0 + 4 = 4,$$

$$U_4 = \min\{U_1 + d_{14}; U_2 + d_{24}; U_3 + d_{34}\} = \min\{0 + 10; 2 + 11; 4 + 3\} = 7,$$

$$U_5 = \min\{U_2 + d_{25}; U_4 + d_{45}\} = \min\{2 + 5; 7 + 8\} = 7,$$

$$U_6 = \min\{U_3 + d_{36}; U_4 + d_{46}\} = \min\{4 + 1; 7 + 7\} = 5,$$

$$U_7 = \min\{U_5 + d_{57}; U_6 + d_{67}\} = \min\{7 + 6; 5 + 9\} = 13.$$

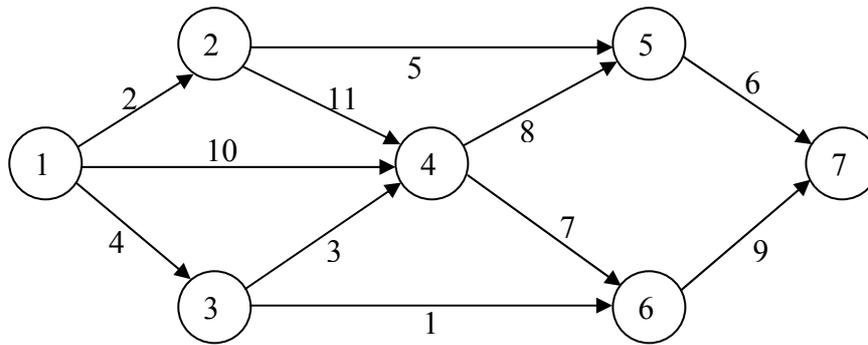


Рис. 3.22

Минимальное расстояние между узлами 1 и 7 равно 13, а соответствующий маршрут: 1 – 2 – 5 – 7.

### Задача замены автомобильного парка

Фирма по прокату автомобилей планирует замену автомобильного парка на очередные 5 лет. Автомобиль должен проработать не менее 1 года, прежде чем фирма поставит вопрос о его замене. На рис. 3.23 приведены стоимости замены автомобилей (усл. ед.), зависящие от времени замены и количества лет, в течение которых автомобиль находился в эксплуатации.

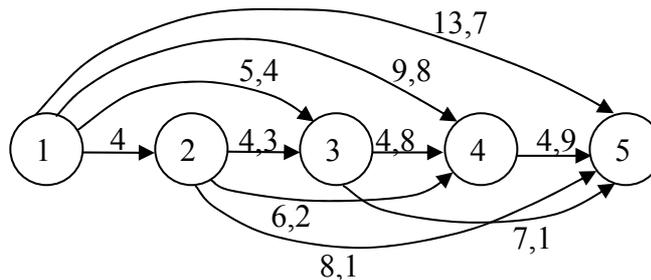


Рис. 3.23

Определить план замены автомобилей, обеспечивающий при этом минимальные расходы.

**РЕШЕНИЕ.** Найдем минимальные расстояния:

$$U_1 = 0,$$

$$U_2 = U_1 + d_{12} = 0 + 4 = 4,$$

$$U_3 = \min\{U_1 + d_{13}; U_2 + d_{23}\} = \min\{0 + 5,4; 4 + 4,3\} = 5,4,$$

$$U_4 = \min\{U_1 + d_{14}; U_2 + d_{24}; U_3 + d_{34}\} = \min\{0 + 9,8; 4 + 6,2; 5,4 + 4,8\} = 9,8,$$

$$U_5 = \min\{U_1 + d_{15}; U_2 + d_{25}; U_3 + d_{35}; U_4 + d_{45}\} = \min\{0 + 13,7; 4 + 8,1; 5,4 + 7,1; 9,8 + 4,9\} = 12,1.$$

Кратчайший путь 1– 2 – 5 со стоимостью 12,1 усл. ед. Это означает, что каждый автомобиль заменяется через 2 года, а через 5 – списывается.

## УПРАЖНЕНИЯ

1. Составить сетевой график выполнения работ и рассчитать временные параметры по данным, представленным в табл. 3.5.

Таблица 3.5

Содержание работы	Обозначение	Предыдущая работа	Продолжительность, дн.
Составление сметы	a <sub>1</sub>		10
Заказ и доставка оборудования	a <sub>2</sub>	a <sub>1</sub>	15
Распределение кадров	a <sub>3</sub>	a <sub>1</sub>	5
Установка оборудования	a <sub>4</sub>	a <sub>2</sub>	20
Подготовка кадров	a <sub>5</sub>	a <sub>3</sub>	9
Оформление торгового зала	a <sub>6</sub>	a <sub>4</sub>	8
Поставка товаров	a <sub>7</sub>	a <sub>5</sub>	7
Заказ и получение ценников	a <sub>8</sub>	a <sub>5</sub>	5
Заказ и получение формы	a <sub>9</sub>	a <sub>5</sub>	6
Выкладка товаров	a <sub>10</sub>	a <sub>6</sub> , a <sub>7</sub>	3
Заполнение ценников	a <sub>11</sub>	a <sub>8</sub>	4
Генеральная репетиция	a <sub>12</sub>	a <sub>9</sub> , a <sub>10</sub> , a <sub>11</sub>	2

2. Постройте график работ, определите критический путь и стоимость работ при нормальном режиме, критический путь и минимальную стоимость работ при максимальном режиме. Исходные данные указаны в табл. 3.6.

3. Постройте график работ, определите критический путь и стоимость работ при нормальном режиме, критический путь и минимальную стоимость работ при максимальном режиме. Необходимые исходные данные приведены в табл. 3.7.

4. Для улучшения финансового состояния фирме необходимо увеличить спрос на выпускаемый цемент марки М400 и расширить потребительский рынок. Фирма считает целесообразным размещать цемент в специализированной таре. Для переоснащения цеха необходимо установить оборудование по производству специализированной тары. Предполагается выполнить следующее:

- 1) подготовку и выпуск технического задания на переоборудование цеха (20 дн.);
- 2) разработку мероприятий по технике безопасности (25 дн.);
- 3) подбор кадров (10 дн.);
- 4) заказ и поставку необходимого оборудования (30 дн.);
- 5) заказ и поставку электрооборудования (40 дн.);

Таблица 3.6

Операция	Нормальный режим работ		Максимальный режим работ	
	Продолжительность, дн.	Стоимость, ден. ед.	Продолжительность, дн.,-	Стоимость, ден. ед.
1, 2	4	80	2	150
1, 3	2	50	1	70
1, 4	3	60	2	80
2, 4	2	60	1	70
2, 6	6	100	3	160
3, 4	2	40	1	60
3, 5	3	70	2	90
4, 6	4	90	2	170
5, 6	4	80	2	160

Таблица 3.7

Операции ij	Нормальный режим работ		Максимальный режим работ	
	Продолжительность, дн.	Стоимость, ден. ед.	Продолжительность, дн.	Стоимость, ден. ед.
1, 2	5	110	4	130
1, 3	3	70	2	90
1, 4	2	50	1	60
2, 5	3	60	2	80
2, 6	4	80	2	110
3, 6	2	60	1	70
4, 7	6	110	4	150
5, 7	3	70	2	80
6, 7	5	100	2	150

- б) установку оборудования (50 дн.);
- 7) установку электрооборудования (45 дн.);
- 8) обучение персонала (15 дн.);
- 9) испытание и сдачу в эксплуатацию линии (25 дн.).

Ожидается, что производительность вводимой линии по производству тары составит 1000 мешков в день при односменном режиме работы. Стоимость 1 мешка – 25 р., выручка от реализации тары в смену составит 25 тыс. р., из которых чистая прибыль фирмы равна 50 тыс. р. Деньги на покупку оборудования и переоснащение цеха в размере 5500 тыс. р. взяты в банке под 30% годовых из расчета 5000 тыс. р. на оборудование и 500 тыс. р. на его установку.

Затраты на проведение работ и их продолжительность в нормальном и максимальном режимах указаны в табл. 3.8.

Таблица 3.8

Работа	Нормальный режим работы		Максимальный режим работы	
	Продолжительность, дн.	Затраты, тыс. р.	Продолжительность, дн.	Затраты, тыс. р.
1	20	20	18	26
2	25	30	20	37
3	10	5	9	7
4	30	60	23	64
5	40	65	32	78
6	50	90	43	100
7	45	80	41	85
8	15	5	9	10
9	25	50	21	57

Составить график проведения работ, определить критический путь и стоимость работ по переоборудованию цеха при нормальном режиме работ.

Провести «сжатие» работ, определить, через какое время после начала выпуска тары фирма может вернуть кредит банку, и минимальную суммарную стоимость работ.

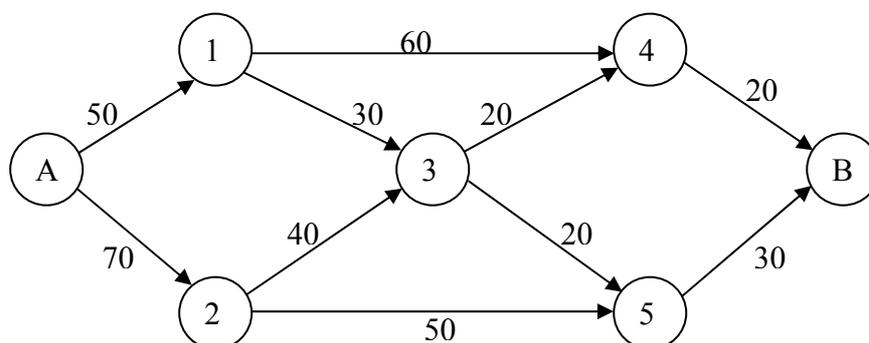


Рис. 3.24

5. Автотранспортному предприятию предстоит освоить новый маршрут между городами *A* и *B*. На рис. 3.24 представлены различные маршруты следования из *A* в *B*, проходящие через несколько других поселков. Расстояния указаны (числами в километрах) около стрелок.

Определить кратчайший маршрут следования автобусов из города *A* в город *B*.

6. Пожарной службе необходимо определить кратчайший путь от гаража (пункт *A*) до нефтеперерабатывающего завода (пункт *B*) по данным в километрах, указанным на рис. 3.25.

7. Строительной фирме необходимо проложить водопроводные трубы к 9 объектам, на которых она ведет строительство.

Числа на ребрах указывают длину труб в метрах. Узел 1 – подсоединение

к водопроводной трассе (рис. 3.26).

Отсутствие ребра между двумя узлами означает, что соединение соответствующих объектов невозможно.

Найти такое соединение узла 1 с объектами строительства, чтобы суммарная длина трубопроводов была минимальной.

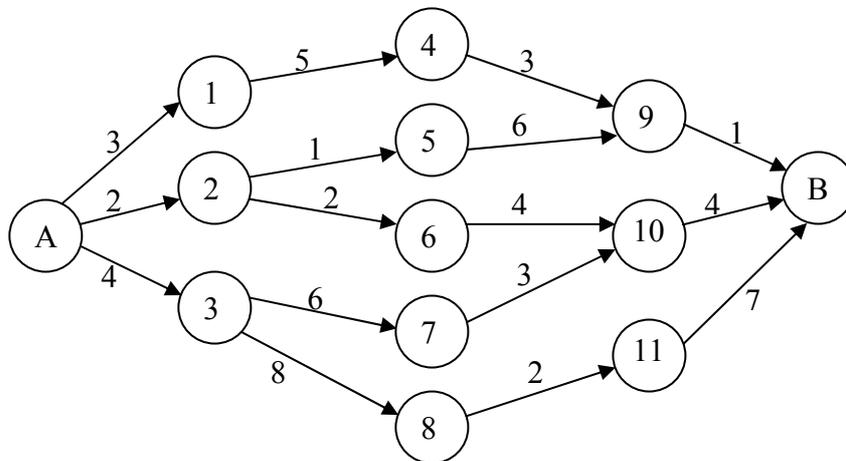


Рис. 3.25

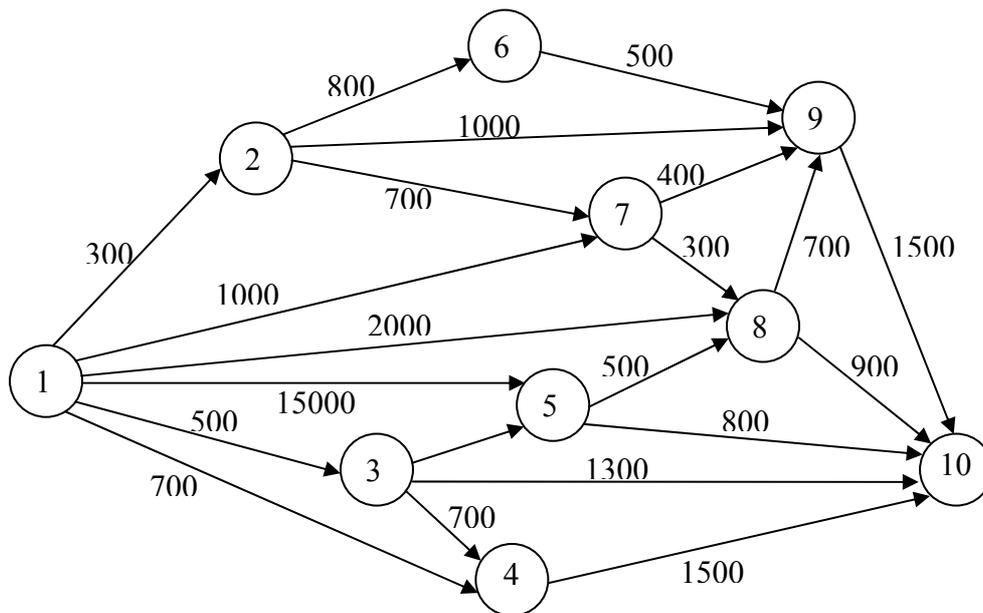


Рис. 3.26

**8.** Фирма по прокату видео– и стерео кассет планирует их замену на очередные 5 лет. Партия кассет должна эксплуатироваться не менее одного года, прежде чем ее заменяют. На рис. 3.27 приведены стоимости замены партии кассет (в тыс. р.), зависящие от времени замены и числа лет, в течение которых кассеты находятся в эксплуатации.

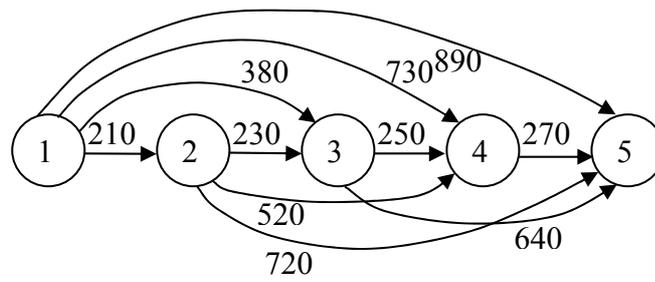


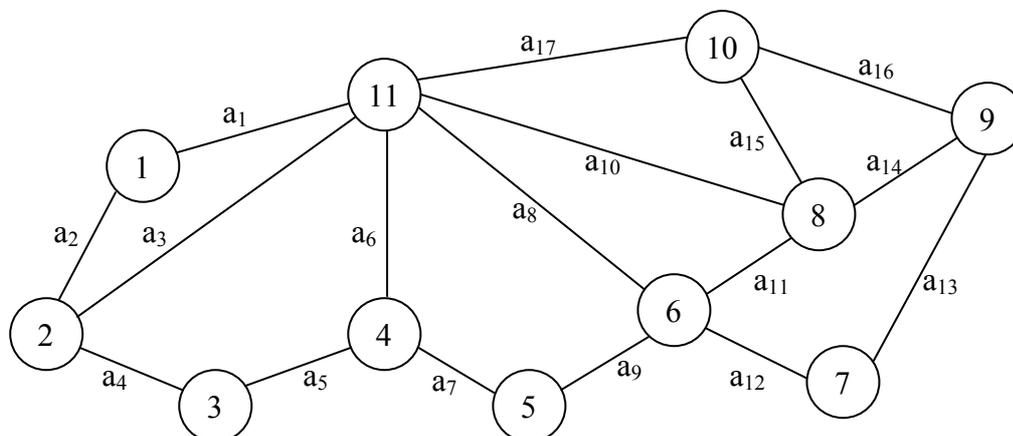
Рис. 3.27

Определить план замены кассет, обеспечивающий фирме минимальные расходы.

## ЗАДАЧИ ДЛЯ САМОСТОЯТЕЛЬНОГО РЕШЕНИЯ

1. Районной администрацией принято решение о газификации одного из небольших сел района, имеющего 10 жилых домов

Расположение домов указано на рисунке. Числа в кружках обозначают условный номер дома. Узел 11 является газопонижающей станцией.



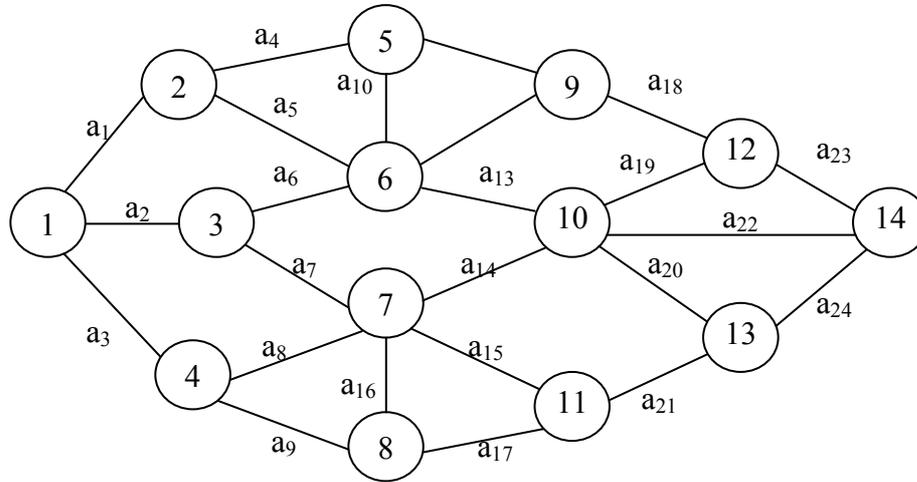
Разработать такой план газификации села, чтобы общая длина трубопровода была наименьшей.

*Значение коэффициентов условия задач*

№ варианта	1	2	3	4	5	6	7	8	9	10
$a_1$	200	180	220	150	170	190	230	160	210	240
$a_2$	60	70	50	40	80	70	30	100	90	40
$a_3$	250	270	290	220	230	240	280	250	260	300
$a_4$	110	130	120	140	100	150	200	170	190	180
$a_5$	150	140	110	100	120	130	160	150	140	110
$a_6$	300	320	310	350	330	360	340	310	290	370
$a_7$	80	90	70	100	60	50	70	40	50	90
$a_8$	350	370	360	390	340	380	330	390	360	400
$a_9$	120	130	140	190	150	180	170	160	140	160
$a_{10}$	400	440	420	430	470	450	410	460	440	470
$a_{11}$	210	190	200	210	220	180	230	170	180	190
$a_{12}$	40	50	30	60	80	70	90	80	50	40
$a_{13}$	120	130	150	120	100	170	160	70	90	110
$a_{14}$	30	40	50	60	30	50	80	70	90	40
$a_{15}$	70	50	40	60	30	80	70	90	40	50
$a_{16}$	20	40	30	50	30	70	20	60	40	50
$a_{17}$	550	580	570	590	530	520	560	630	600	610

2. Транспортному предприятию требуется перевезти груз из пункта 1 в пункт 14. на рисунке показана сеть дорог и стоимость перевозки единицы груза между отдельными пунктами.

Определить маршрут доставки груза, которому соответствуют наименьшие затраты.



*Значения коэффициентов условия задачи*

№ варианта	1	2	3	4	5	6	7	8	9	10
<b>Значения</b>										
a <sub>1</sub>	20	18	22	15	17	19	23	16	21	24
a <sub>2</sub>	18	19	21	16	18	21	20	15	19	22
a <sub>3</sub>	19	17	20	17	16	20	22	17	20	23
a <sub>4</sub>	11	13	12	14	10	15	20	17	19	18
a <sub>5</sub>	15	14	11	10	12	13	16	15	16	17
a <sub>6</sub>	13	15	10	12	13	16	17	16	18	16
a <sub>7</sub>	12	16	9	11	9	14	19	14	15	19
a <sub>8</sub>	14	17	13	13	11	18	18	19	17	20
a <sub>9</sub>	12	18	14	16	15	17	15	18	14	21
a <sub>10</sub>	24	21	20	18	17	16	19	16	22	23
a <sub>11</sub>	21	19	20	21	22	18	23	17	18	19
a <sub>12</sub>	20	22	19	23	18	17	24	16	20	21
a <sub>13</sub>	22	21	18	22	21	19	20	18	19	18
a <sub>14</sub>	23	23	21	20	19	16	22	15	21	20
a <sub>15</sub>	24	18	17	24	20	15	21	19	22	22
a <sub>16</sub>	20	21	23	19	22	18	20	16	17	21
a <sub>17</sub>	22	17	19	23	18	17	19	22	20	21
a <sub>18</sub>	31	32	30	35	37	36	33	36	31	34
a <sub>19</sub>	32	33	29	31	36	37	34	35	32	33
a <sub>20</sub>	35	37	32	33	34	38	36	31	36	30
a <sub>21</sub>	37	36	31	34	36	35	40	37	39	38
a <sub>22</sub>	45	41	43	42	44	40	46	45	47	45
a <sub>23</sub>	28	32	30	25	26	28	33	31	29	27
a <sub>24</sub>	30	31	32	24	25	29	32	33	30	29

3. Составить сетевой график выполнения работ и рассчитать временные параметры по данным, представленным в таблице.

Содержание работы	Обозначение	Предыдущая работа	Продолжительность, дн.
Исходные данные на изделие	$a_1$		$t_1$
Заказ комплектующих деталей	$a_2$	$a_1$	$t_2$
Выпуск документации	$a_3$	$a_1$	$t_3$
Изготовление деталей	$a_4$	$a_3$	$t_4$
Постановка комплектующих деталей	$a_5$	$a_2$	$t_5$
Сборка изделия	$a_6$	$a_4, a_5$	$t_6$
Выпуск документации на испытание	$a_7$	$a_3$	$t_7$
Испытание и приемка изделия	$a_8$	$a_6, a_7$	$t_8$

*Значения коэффициентов условия задачи*

№ варианта	1	2	3	4	5	6	7	8	9	10
<b>Значения</b>										
$t_1$	30	33	36	35	25	20	15	30	25	20
$t_2$	7	9	8	6	8	11	10	5	9	7
$t_3$	15	17	18	14	16	20	12	13	20	19
$t_4$	35	33	32	34	31	35	30	37	39	38
$t_5$	25	24	21	20	22	23	26	25	18	21
$t_6$	13	15	10	12	13	16	17	16	18	16
$t_7$	12	16	9	11	9	14	19	14	15	19
$t_8$	14	17	13	13	11	18	18	19	17	20

4. Постройте график работ, определите критический путь и стоимость работ до сжатия. Найдите критический путь и минимальную стоимость работ после сжатия.

Операция	Нормальный режим		Максимальный режим	
	Продолжительность, дн.	Затраты, тыс. р.	Продолжительность, дн.	Затраты, тыс. р.
1, 2	$a_{11}$	$b_{11}$	$a_{12}$	$b_{12}$
2, 3	$a_{21}$	$b_{21}$	$a_{22}$	$b_{22}$
2, 4	$a_{31}$	$b_{31}$	$a_{32}$	$b_{32}$
2, 5	$a_{41}$	$b_{41}$	$a_{42}$	$b_{42}$
3, 5	$a_{51}$	$b_{51}$	$a_{52}$	$b_{52}$
4, 5	$a_{61}$	$b_{61}$	$a_{62}$	$b_{62}$
5, 6	$a_{71}$	$b_{71}$	$a_{72}$	$b_{72}$

*Значения коэффициентов условия задачи*

<b>№ варианта</b>	<b>1</b>	<b>2</b>	<b>3</b>	<b>4</b>	<b>5</b>	<b>6</b>	<b>7</b>	<b>8</b>	<b>9</b>	<b>10</b>
<b>Значения</b>										
a <sub>11</sub>	20	18	22	15	17	19	23	16	21	24
a <sub>21</sub>	28	29	31	36	28	31	30	25	29	32
a <sub>31</sub>	19	17	20	17	16	20	22	17	20	23
a <sub>41</sub>	21	23	22	24	20	25	30	27	29	28
a <sub>51</sub>	25	24	21	20	22	23	26	25	26	27
a <sub>61</sub>	13	15	10	12	13	16	17	16	18	16
a <sub>71</sub>	12	16	9	11	9	14	19	14	15	19
b <sub>11</sub>	14	17	13	13	11	18	18	19	17	20
b <sub>21</sub>	12	18	14	16	15	17	15	18	14	21
b <sub>31</sub>	24	21	20	18	17	16	19	16	22	23
b <sub>41</sub>	21	19	20	21	22	18	23	17	18	19
b <sub>51</sub>	20	22	19	23	18	17	24	16	20	21
b <sub>61</sub>	22	21	18	22	21	19	20	18	19	18
b <sub>71</sub>	23	23	21	20	19	16	22	15	21	20
a <sub>12</sub>	17	15	18	13	14	15	18	13	16	18
a <sub>22</sub>	23	24	24	30	22	26	24	21	25	27
a <sub>32</sub>	16	14	16	15	13	17	18	22	16	19
a <sub>42</sub>	17	20	18	21	16	21	25	22	24	24
a <sub>52</sub>	20	21	18	17	18	19	21	21	22	23
a <sub>62</sub>	11	13	8	10	11	12	14	15	14	13
a <sub>72</sub>	10	13	8	9	7	11	15	12	12	16
b <sub>12</sub>	17	21	15	16	14	23	24	25	22	25
b <sub>22</sub>	15	22	17	20	19	22	21	23	18	27
b <sub>32</sub>	30	26	25	23	24	20	25	21	27	29
b <sub>42</sub>	26	25	24	27	28	23	29	22	23	24
b <sub>52</sub>	25	28	14	29	22	23	30	20	27	29
b <sub>62</sub>	28	27	15	29	26	15	26	24	23	22
b <sub>72</sub>	29	30	26	25	24	21	29	20	27	26

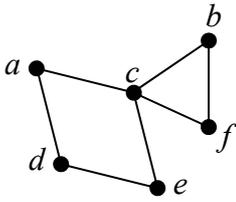
## ОТВЕТЫ К УПРАЖНЕНИЯМ

### Раздел 1.

1. (а).
3. (в).
5. (а), (в).
7. (б).
9. Предположим, что граф  $G$  связный и ровно две его вершины имеют нечетную степень. Пусть  $a$  и  $b$  – вершины нечетной степени. Если между  $a$  и  $b$  ребро отсутствует, добавим его. Теперь каждая вершина имеет четную степень, поэтому новый граф имеет эйлеров цикл. В этом цикле следует проходить и ребро  $\{a, b\}$ . Например, проходя цикл, мы движемся от  $a$  к  $b$ . Если начать с этого ребра и пройти цикл, то легко заметить, что если ребро удалить, начать путь в вершине  $b$  и пройти его, следуя циклу, то получим эйлеров путь из  $b$  в  $a$ . Если между  $a$  и  $b$  имеется ребро, удалим его. Новый граф имеет эйлеров цикл, если он по-прежнему связный. Пусть эйлеров цикл начинается и заканчивается в вершине  $a$ . Если мы пройдем этот цикл, а затем проследуем вдоль удаленного ребра от  $a$  к  $b$ , то получим эйлеров путь от  $a$  к  $b$ . Если новый граф перестал быть связным, то он имеет эйлеров цикл для компоненты, содержащей вершину  $a$ , который начинается и заканчивается в вершине  $b$ . Пройдем эйлеров цикл от  $a$  к  $a$ , удаленное ребро от  $a$  к  $b$ , затем пройдем эйлеров цикл от  $b$  к  $b$ . В результате получим эйлеров путь от  $a$  к  $b$ .  
Предположим, что граф  $G$  имеет эйлеров путь. Пусть, например, он начинается в вершине  $a$  и заканчивается в вершине  $b$ . После первого ребра пути, выходящего из  $a$ , для каждого ребра пути, который ведет в  $a$ , должно существовать ребро, выходящее из  $a$ . Поэтому вершина  $a$  должна иметь нечетную степень. Аналогично, вершина  $b$  должна иметь нечетную степень. В любых других вершинах, для любого ребра пути, который ведет в эту вершину, должно существовать ребро, которое выходит из этой вершины. Поэтому вершина имеет четную степень.
11. Если граф сильно связный, то для любой его вершины  $v$ , несомненно, любая другая вершина достижима из вершины  $v$ , и  $v$  достижима из любой другой его вершины. Обратно, пусть имеется вершина  $v$ , обладающая указанным свойством. Пусть  $a$  и  $b$  – произвольные вершины. Поскольку существует путь из  $a$  в  $v$  и путь из  $v$  в  $b$ , то существует путь из  $a$  в  $b$ , поэтому рассматриваемый граф сильно связный.
13. а)  $acg fedba$ ; б)  $abfecda$ ; в)  $abihfdcega$ ; г) не существует; д) не существует.
15. а)  $abcde$ ; б)  $abcdef$ ; в)  $fdcegabih$  г)  $egacdfhb$  д)  $adbcefgh$ .
17. Если граф  $G$  имеет гамильтонов цикл, то любое ребро является частью цикла. Но разрезающее ребро не может быть ребром, входящим в цикл. Следовательно,  $G$  не содержит разрезающее ребро. Путь  $\{a, b\}$  – разрезающее ребро и компоненты  $C_1$  и  $C_2$  – компоненты, имеющие гамильтоновы циклы, на-

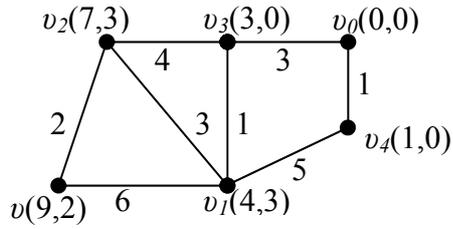
пример,  $av_1v_2v_3 \dots v_{k-1}a$  и  $bv_1'v_2'v_3' \dots v_{m-1}'b$ . Тогда  $v_1v_2v_3 \dots v_{k-1}abv_1'v_2'v_3' \dots v_{m-1}'$  — ГАМИЛЬТОНОВ ПУТЬ.

19.

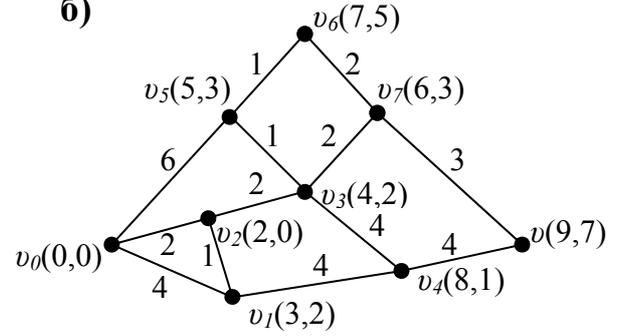


20.

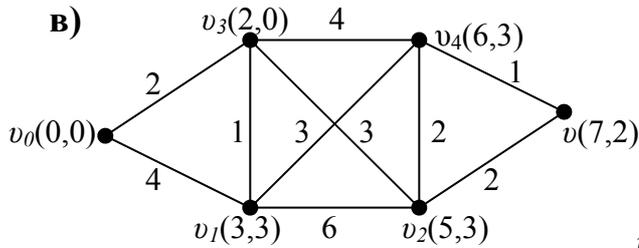
а)



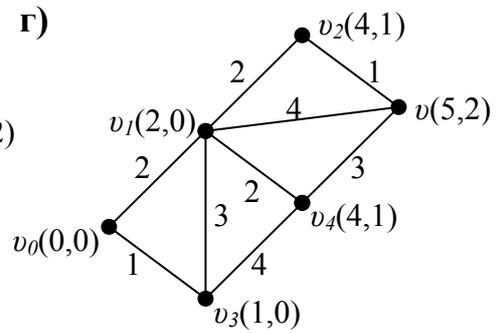
б)



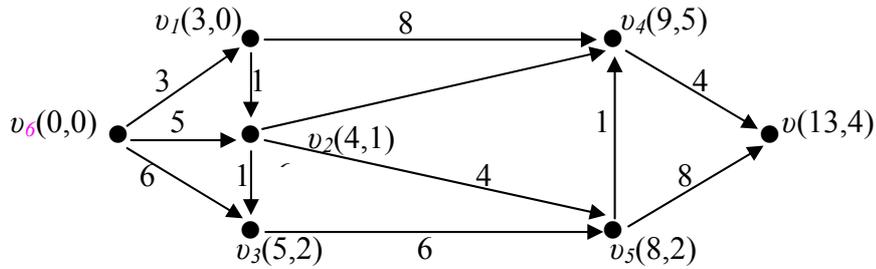
в)



г)

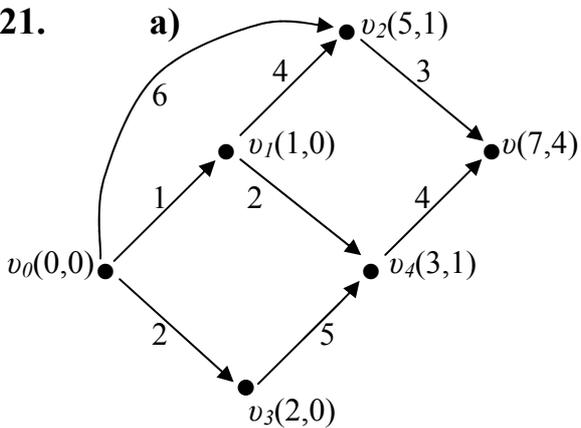


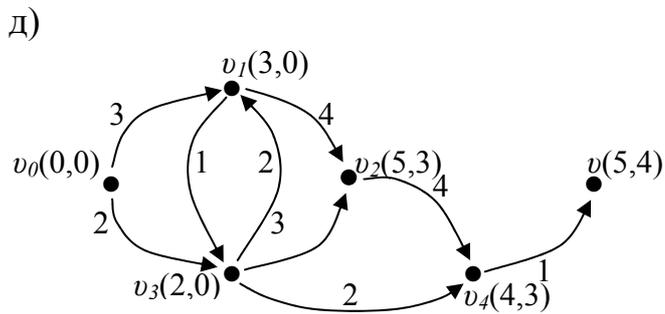
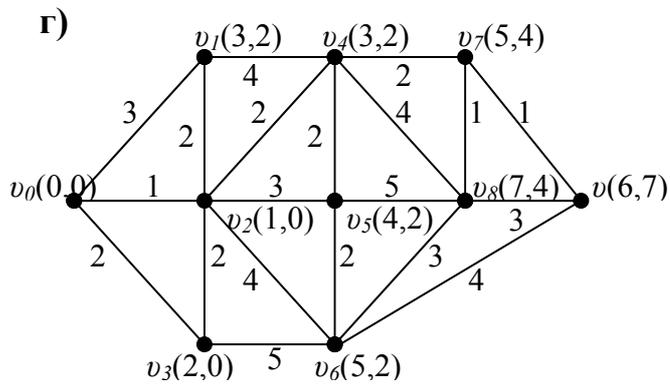
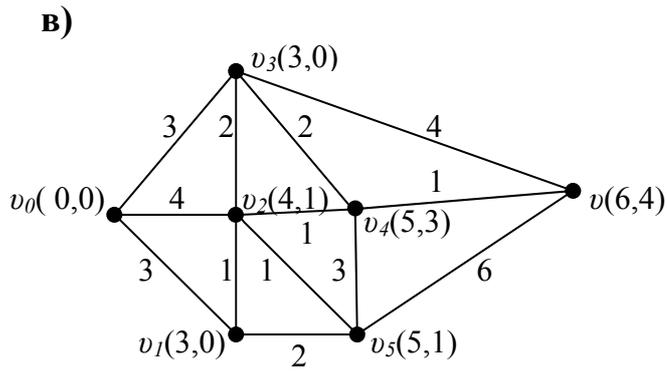
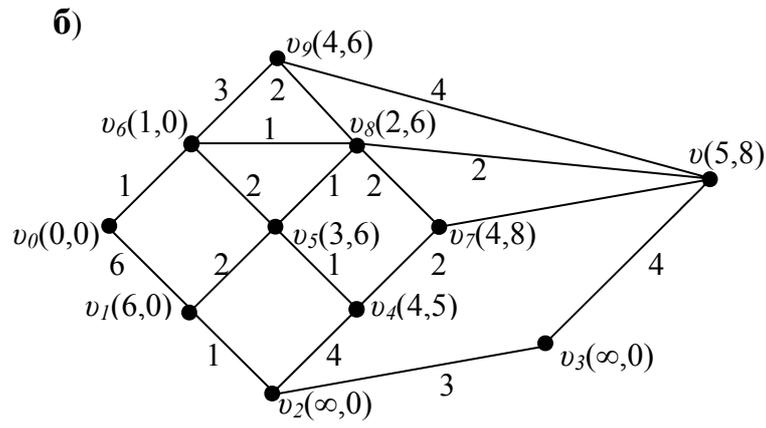
г)



21.

а)





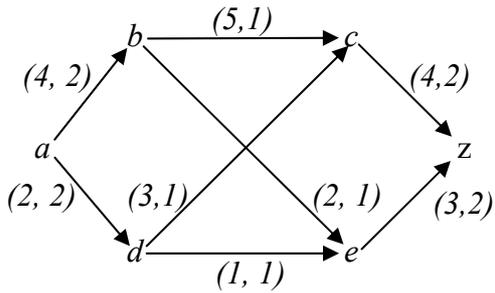
## Раздел 2

1. а)

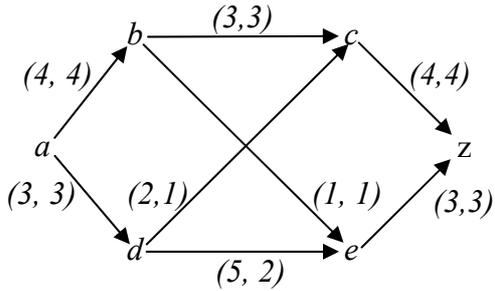
вершина	вход.поток	ребро	выход.поток	ребро
b	2	(a, b)	3	(b, c)
	1	(e, b)		
c	3	(b, c)	2	(c, z)
			1	(c, d)
d	3	(a, d)	4	(d, e)
	1	(c, d)		

б) 5;      в) 10;      г) 10;      д) 11.

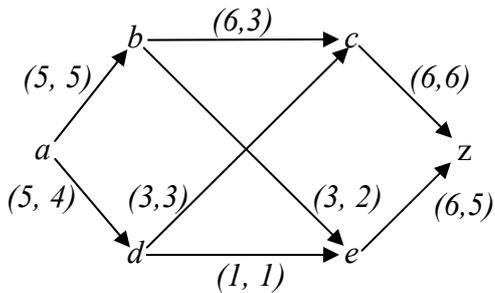
3.



5.



7.



9. {a}.

11. {a, d}

13. (i) (a), (б), (в);

(ii) (б), (в), (е);

(iii) (a), (в), (г), (е), (ж);

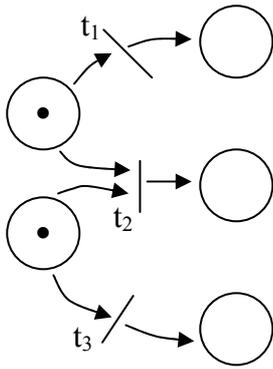
(iv) (a), (б), (в), (г), (д), (е), (ж);

(v) (a), (в), (г), (е), (ж);

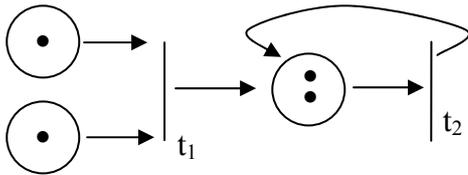
(vi) (б), (в), (г), (д), (е), (ж);

(vii) (a), (б), (в), (г), (д), (е), (ж).

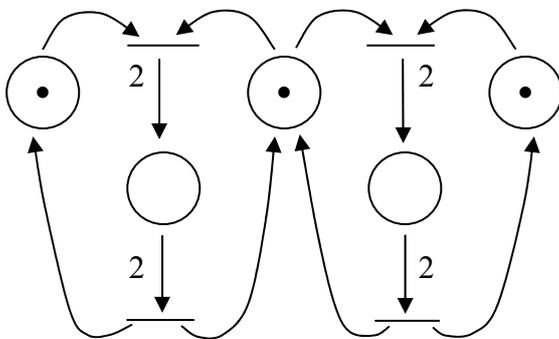
15.



17.

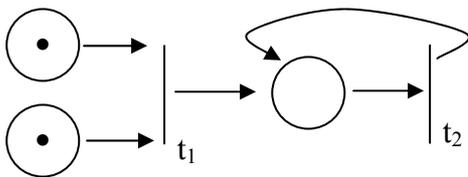


19.



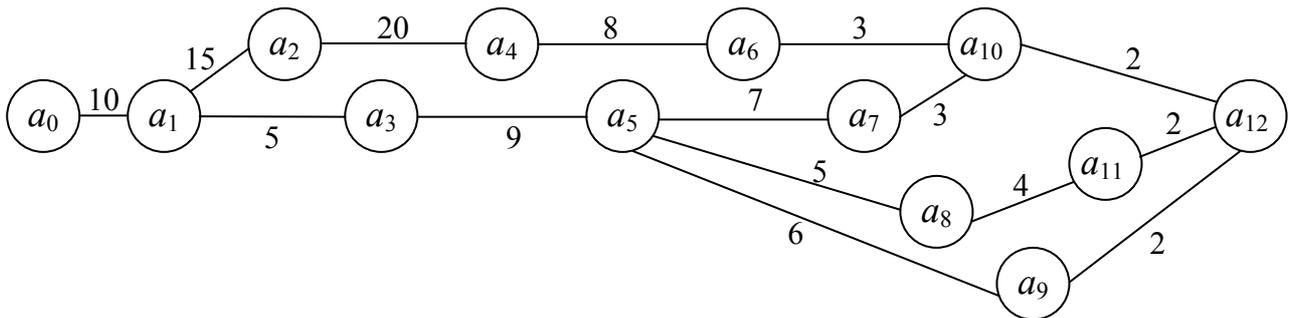
21. Да

23.



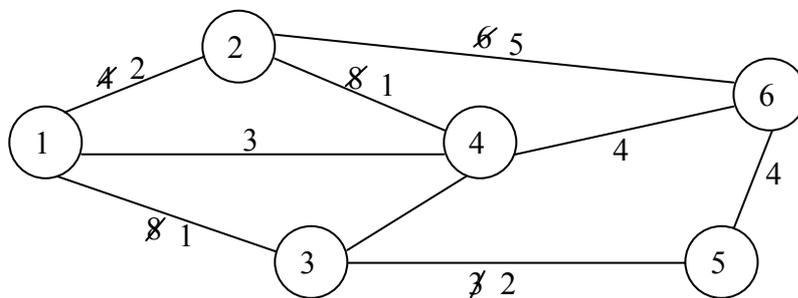
### РАЗДЕЛ 3.

1. Критический путь  $a_0 - a_1 - a_2 - a_4 - a_6 - a_{10} - a_{12}$  равен 58 дн.



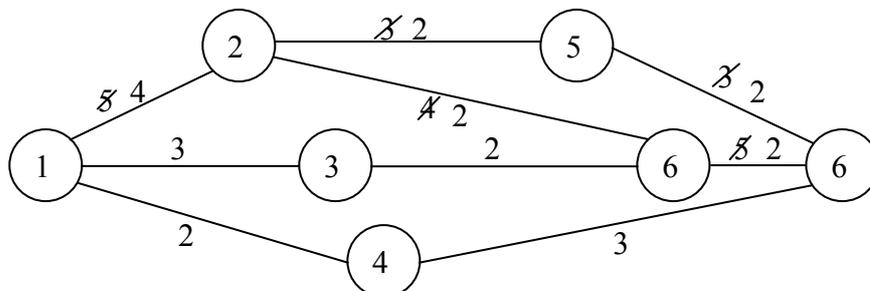
2. При нормальном режиме работ 2 критических пути: 1 – 2 – 6 и 1 – 2 – 4 – 6, длина пути 10 дн., стоимость работ 630 ден. ед.

При максимальном режиме работ 5 критических путей: 1 – 2 – 6, 1 – 2 – 4 – 6, 1 – 4 – 6, 1 – 3 – 4 – 6, 1 – 3 – 5 – 6, длина пути 7 дн., минимальная стоимость работ 770 ден. ед.



3. При нормальном режиме работ 1 критический путь 1 – 2 – 6 – 7, длина пути 14 дн., стоимость работ 700 ден. ед.

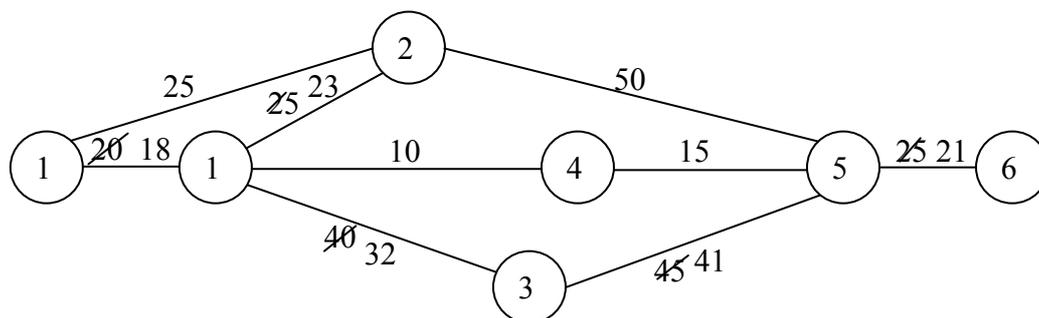
При максимальном режиме работ 3 критических пути: 1 – 2 – 6 – 7, 1 – 2 – 5 – 7, 1 – 4 – 7, длина пути 8 дн., минимальная стоимость работ 820 ден. ед.



Предприятие может вернуть кредит банку через 208 дней.

4. При нормальном режиме работ 1 критический путь 0 – 1 – 3 – 5 – 6, длина пути 130 дн., стоимость работ 405 млн. р.

При максимальном режиме работ 2 критических пути: 0 – 1 – 2 – 5 – 6, 0 – 1 – 3 – 5 – 6, длина пути 112 дн., минимальная стоимость работ 438,8 ден. ед.



5. Кратчайший путь А – 1 – 3 – 4 – В длиной 120.

6. Кратчайший путь А – 2 – 5 – 9 – В длиной 10.

7. Минимальная длина 1800 м.

8. Каждая партия кассет должна заменяться через 1 год, а через 5 лет списываться.

## ЗАКЛЮЧЕНИЕ

На основании вышеизложенного теоретического и практического материала можно сделать следующие выводы:

– теория графов, как один из основных разделов классической высшей математики, позволяет развивать у студентов пространственное аналитическое мышление, что способствует более четкому восприятию экономических процессов и их представлению в виде пространственных графоаналитических моделей.

– сетевое моделирование как прообраз будущего процесса также строится на основе знаний теорий графов, т.е. изображений графоаналитическим методом процессов построения, что позволяет уяснить студенту полноту и последовательность процесса прогнозирования. С расчетом всех необходимых технико-экономических параметров определяется длительность проектного и производственного цикла, трудоемкость общего процесса и других планово-экономических показателей процесса.

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Андерсон, Джеймс А. Дискретная математика и комбинаторика : пер. с англ. / Джеймс А. Андерсон. – М. : Издательский дом «Вильямс», 2003. – ... с.
2. Белоусов, А. И. Дискретная математика / А. И. Белоусов, С. Б. Ткачев; под редакцией В. С. Зарубина, А. П. Крищенко. – 2004. – ... с.
3. Конюховский, П. В. Математические методы исследования операций в экономике / П. В. Конюховский. – СПб. : Питер, 2000. – ... с.
4. Красс, М. С. Основы математики и ее приложения в экономическом образовании : учебник / М. С. Красс, Б. П. Чупрынов. – 2-е изд., испр. – М. : Дело, 2001. – ... с.
5. Новиков, Ф. А. Дискретная математика для программистов : учебник для вузов / Ф. А. Новиков. – 2-е изд. – СПб. : Питер, ... . – ... с.

Учебное издание

ХАРИТОНОВА Елена Викторовна

**ГРАФЫ И СЕТИ**

Учебное пособие

Корректор *А. В. Власова*

Подписано в печать 03.06.2006. Формат 60×84/16.

Бумага офсетная. Печать трафаретная. Усл. печ. л. 5,29.

Тираж 100 экз. Заказ .

Ульяновский государственный технический университет  
432027, г. Ульяновск, ул. Сев. Венец, 32  
Типография УлГТУ, 432027, г. Ульяновск, ул. Сев. Венец, 32

## БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Андерсон, Джеймс А. Дискретная математика и комбинаторика : пер. с англ. / Джеймс А. Андерсон. – М. : Издательский дом «Вильямс», 2003. – 960 с.
2. Белоусов, А. И. Дискретная математика / А. И. Белоусов, С. Б. Ткачев; под редакцией В. С. Зарубина, А. П. Крищенко. – М. : Издательство МГТУ им. Н. Э. Баумана, 2004. – 744 с.
3. Конюховский, П. В. Математические методы исследования операций в экономике / П. В. Конюховский. – СПб. : Питер, 2000. – 208 с.
4. Красс, М. С. Основы математики и ее приложения в экономическом образовании : учебник / М. С. Красс, Б. П. Чупрынов. – 2-е изд., испр. – М. : Дело, 2001. – 688 с.
5. Новиков, Ф. А. Дискретная математика для программистов : учебник для вузов / Ф. А. Новиков. – 2-е изд. – СПб. : Питер, 2005. – 364 с.

## РЕЦЕНЗИЯ

на учебное пособие «Графы и сети»  
старшего преподавателя кафедры Общенаучных дисциплин  
ИАТУ УлГТУ Харитоновой Е.В.

Рецензируемое учебное пособие предназначено для студентов специальности 080109 «Бух.учет, анализ и аудит». Пособие состоит из введения, трех разделов, задач для самостоятельного решения, ответов к упражнениям и библиографического списка используемой литературы.

Во введении автор описывает область применения теории графов и сетей.

В первых двух разделах пособия «Графы», «Сети и потоки» в достаточном объеме рассмотрен теоретический материал. При изложении материала, автор придерживается принципов научности, системности и доступности. Используются точные правильные определения терминов и точные формулировки теорем. В содержании разделов приведено достаточное количество примеров и иллюстраций, что способствует эффективному использованию учебного пособия при организации самостоятельной работы студентов, при подготовке к практическим занятиям и экзаменам.

Третий раздел пособия посвящен основным типам задач по рассматриваемой тематике и способам их решения. В этом разделе подробно и методически грамотно изложено решение типовых задач, что поможет студентам усвоить теоретический материал и научиться решать задачи.

Наличие задач для самостоятельного решения (10 вариантов) позволит преподавателю организовать проверку знаний студентов по теме «Сети».

Ответы к упражнениям помогут студентам своевременно контролировать правильность решения и устранять допущенные ошибки.

В целом учебное пособие Харитоновой Е.В. «Графы и сети» написано в соответствии с учебной программой дисциплины «Математика» специальности 080109 «Бух.учет, анализ и аудит» и отвечает требованиям государственного образовательного стандарта по специальности.

Считаю, что рукопись Харитоновой Е.В. «Графы и сети» заслуживает опубликования в представленном виде, а учебное пособие может быть использовано в учебном процессе.

Кандидат физико-математических наук,  
доцент кафедры «Экономико-математические  
методы и информационные технологии»  
Института экономики и бизнеса УлГУ

**А.Е. Эткин**