

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ
БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ
УНИВЕРСИТЕТ имени академика С.П.КОРОЛЕВА
(НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ)» (СГАУ)

ОРИЕНТИРОВАННЫЕ И НЕОРИЕНТИРОВАННЫЕ ГРАФЫ В ПРИМЕРАХ И ЗАДАЧАХ

Рекомендовано редакционно-издательским советом
федерального государственного бюджетного образовательного
учреждения высшего профессионального образования
«Самарский государственный аэрокосмический
университет имени академика С.П.Королева
(национальный исследовательский университет)»
в качестве методических указаний

Самара
Издательство СГАУ
2014

УДК СГАУ: 5(075)

Составители: *О. Л. Старинова, О.А. Васильева, Е.В. Барина*

Рецензент В. Д. Е л е н е в

Ориентированные и неориентированные графы в примерах и задачах: метод. указания / сост.: *О. Л. Старинова, О.А. Васильева, Е.В. Барина*. – Самара: Изд-во СГАУ, 2014. – 56 с.

Методические указания составлены в соответствии с действующей программой по курсу высшей математики для специальностей 160400 «Проектирование, производство и эксплуатация ракет и ракетно-космических комплексов», 160400 «Ракетные комплексы и космонавтика» СГАУ. Методические указания содержат 30 вариантов заданий по теории графов с подробно разобранными примерами для каждого задания.

Методические указания могут быть рекомендованы студентам для выполнения расчётно-графической работы по теории графов.

СОДЕРЖАНИЕ

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ	4
2. НЕОРИЕНТИРОВАННЫЕ ГРАФЫ	8
2.1 Задание 1.....	8
2.2 Задание 2.....	11
2.3 Задание 3.....	18
3. ОРИЕНТИРОВАННЫЕ ГРАФЫ.....	34
3.1 Задание 1.....	34
3.2 Задание 2.....	39
3.3 Задание 3.....	46
3.4 Задание 4.....	51
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	55

1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Пусть V – непустое множество, $V(2)$ – множество всех его двухэлементных подмножеств. Пара (V, E) , где E – произвольное подмножество множества $V(2)$, называется *графом* (неориентированным графом). Элементы множества V называются *вершинами* графа, а элементы множества E – *рёбрами*. Итак, граф – это конечное множество V вершин и множество E рёбер, $E \subset V(2)$.

Если направление рёбер не указано, то граф называется *неориентированным*. Если направление рёбер указано, то граф называется *ориентированным*, а сами рёбра принято называть дугами. Если в графе имеются и рёбра и дуги, то его называют *смешанным*. Примером неориентированного графа является карта дорог. Примером ориентированного графа – река с притоками. Примером смешанного графа – карта дорог, где часть дорог с односторонним движением.

Ребро называется *петлей*, если его началом и концом является одна вершина. Граф, имеющий петли, называется *псевдографом*.

Две вершины называются *смежными*, если существует соединяющая их дуга.

Ребро e_i называется *инцидентным* вершине v_j , если оно выходит из вершины или входит в неё.

Рёбра (дуги), инцидентные одной паре вершин, называются *кратными*. Граф, имеющий кратные рёбра или дуги, называется *мультиграфом*.

Степень (валентность) вершины называется число инцидентных ей рёбер. *Кратностью пары вершин* называется число соединяющих их рёбер или дуг.

Матрицей смежности неориентированного графа, содержащего n вершин, называется квадратная матрица $A_{n \times n}$, каждый элемент которой a_{ij} определяется следующим образом:

$$a_{ij} = \begin{cases} 1, & \text{если вершины } i \text{ и } j \text{ соединены ребром;} \\ 0, & \text{в противном случае.} \end{cases}$$

Для графа с кратными рёбрами вместо 1 записывается число рёбер между вершинами i и j .

Матрицей инцидентности неориентированного графа с n вершинами и m рёбрами называется матрица $B_{n \times m}$ с n строками и m столбцами, элемент b_{ij} которой определяется следующим образом:

$$b_{ij} = \begin{cases} 1, & \text{если вершина } i \text{ инцидентна ребру } j; \\ 2, & \text{если вершина } i \text{ инцидентна петле } j; \\ 0, & \text{если вершина } i \text{ и ребро } j \text{ не инцидентны.} \end{cases}$$

Матрицей смежности ориентированного графа, содержащего n -вершин, называется квадратная матрица $A_{n \times n}$, каждый элемент которой a_{ij} определяется следующим образом:

$$a_{ij} = \begin{cases} 1, & \text{если дуга выходит из вершины } i \text{ и входит в вершину } j; \\ 0, & \text{в противном случае.} \end{cases}$$

Аналогично для графа с кратными дугами вместо 1 записывается число рёбер, выходящих из вершины i в j .

Матрицей инцидентности ориентированного графа с n вершинами и m рёбрами называется матрица $B_{n \times m}$ с n строками и m столбцами, элемент которой b_{ij} определяется следующим образом:

$$b_{ij} = \begin{cases} 1, & \text{если вершина } i \text{ является началом ребра } j, \\ -1, & \text{если вершина } i \text{ является концом ребра } j, \\ 2, & \text{если вершина } i \text{ инцидентна петле } j, \\ 0, & \text{если вершина и ребро } j \text{ не инцидентны.} \end{cases}$$

Подграфом G_a графа G называется граф, полученный отбрасыванием части вершин и рёбер (дуг) графа G .

Путь в графе G называется такая последовательность дуг, в которой конец каждой последующей дуги совпадает с началом предыдущей. Путь может быть конечным и бесконечным. Путь, в котором никакая дуга не встречается дважды, называется *элементарным* или *простым*. Замкнутая простая цепь называется *циклом*.

В неориентированном графе понятия дуга, путь заменяются соответственно на ребро, цепь. *Ребро* – отрезок, соединяющий две вершины, *цепь* – последовательность рёбер.

Граф называется *связанным*, если любые его две вершины можно соединить цепью. Граф *сильно связан*, если для его двух любых вершин $x_i \neq x_j$ существует путь, идущий из x_i в x_j .

Граф, который не является связанным, может быть разбит на конечное число связных графов, называемых *компонентами* или *частями*.

Ребро графа G называется *мостом*, если граф, полученный из G путем удаления этого ребра, имеет больше компонент связности, чем граф G .

Точкой сочленения графа называется вершина, удаление которой приводит к увеличению числа его компонент связности.

Дерево – это конечный, связный, неориентированный граф, не имеющий циклов. Характеристическое свойство деревьев состоит в том, что любые две вершины дерева соединены единственной цепью.

Граф называется *планарным*, если он может быть изображен на плоскости таким образом, что его рёбра будут пересекаться только в планарных вершинах.

Эйлеровым путем в графе G называется такой путь, в котором каждое ребро встречается один раз. Эйлер доказал, что такой путь существует тогда и только тогда, когда граф G содержит не более двух вершин нечётной степени и является связным. Если граф содержит точно две вершины нечётной степени, то в эйлеровом пути эти вершины должны быть конечными. Если вершин нечётной степени нет, то граф имеет замкнутый эйлеров путь (эйлеров цикл) и называется *эйлеровым*. Если эйлеров путь не замкнут, граф называется *полуэйлеровым*.

Ориентированный граф содержит эйлеров цикл тогда и только тогда, когда он сильно связан и для каждой вершины графа полустепень захода равна полустепени исхода.

Гамильтоновой цепью называется цепь, содержащая каждую вершину графа ровно один раз. Гамильтонова цепь, начальная и конечная вершины которой совпадают, называется *гамильтоновым циклом*. Граф, содержащий гамильтонов цикл, называется *гамильтоновым*. Если гамильтонов путь не замкнут, граф называется *полугамильтоновым*.

При решении практических задач с применением графов возникает необходимость в разбиении множества вершин графа на классы попарно несмежных между собой вершин. Довольно часто дополнительно требуется, чтобы таких классов было наименьшее число. В теории графов подобные задачи формулируются в терминах раскраски вершин графа.

Пусть $G=(V,E)$ – обыкновенный граф и k – натуральное число. Функция $f: V \rightarrow \{1, \dots, k\}$ называется *раскраской* графа. Раскраска называется *правильной*, если для любых смежных вершин $x, y \in V$ справедливо неравенство $f(x) \neq f(y)$. Число k – *число красок* раскраски f . Можно представить себе, что вершины раскрашиваются некоторым набором красок (в количестве k штук). Правильность раскраски означает, что смежные вершины раскрашиваются в разные цвета.

Наименьшее число красок, необходимое для правильной раскраски графа G , называется *хроматическим числом* графа G . Хроматическое число обозначается через $\chi(G)$.

Рёберной раскраской называется функция $g: E \rightarrow \{1, 2, \dots, k\}$, где E – множество рёбер, k – натуральное число (число красок). Раскраска называется *правильной*, если для любых двух смежных рёбер e_1 и e_2 выполняется неравенство $g(e_1) \neq g(e_2)$. Наименьшее число красок, необходимое для правильной рёберной раскраски графа, называется *хроматическим индексом*.

Взвешенный граф – граф, каждому ребру которого сопоставлено какое-либо значение $\alpha(e)$. Если e – дуга графа G , то число $\alpha(e)$ в

зависимости от контекста называется либо *длиной*, либо *весом*, либо *пропускной способностью дуги*.

Введем *длину пути* как сумму длин дуг, составляющих путь.

Расстоянием $d(u,v)$ от вершины u до вершины v называется длина кратчайшего (u,v) -пути. Если (u,v) -пути не существует, то полагаем $d(u,v)=\infty$. Расстояние от u до u равно нулю. Для случая невзвешенного графа *длиной пути* называют число входящих в этот путь дуг.

Сетью называется пара (G, α) , где G – ориентированный граф, а α – функция из множества рёбер графа в множество неотрицательных действительных чисел. При изображении сети $\alpha(e)$ проставляется рядом с дугой e .

Полустепенью исхода $\rho^-(v)$ вершины v назовем сумму $\alpha(e)$ по всем дугам, исходящим из v , *полустепенью захода* $\rho^+(v)$ вершины v – сумму $\alpha(e)$ по всем дугам, заходящим в v .

Вершину сети v , для которой $\rho^-(v)>0$ и $\rho^+(v)=0$, будем называть *источником*, а вершину w , для которой $\rho^+(w)>0$ и $\rho^-(w)=0$, – *стоком*.

Пусть $N=(G, \alpha)$ – сеть, a – источник и b – сток этой сети. *Потоком через сеть* N называется функция $\varphi: E \rightarrow R^+ \cup \{0\}$, удовлетворяющая двум условиям:

- 1) $\varphi(e) \leq \alpha(e)$ для любой дуги $e \in E$,
- 2) в сети (G, φ) у любой вершины, кроме источника и стока, полустепень исхода равна полустепени захода.

Число $\varphi(e)$ называется *значением потока* через дугу

Поток называется *максимальным*, если он имеет наибольшую величину (среди всех потоков через данную сеть).

2. НЕОРИЕНТИРОВАННЫЕ ГРАФЫ

Задание 1. Дана симметричная матрица Q смежности. Нарисовать на плоскости мультиграф $G=(V,E)$ (единственный с точностью до изоморфизма), имеющий заданную матрицу Q своей матрицей смежности. Найти матрицу инцидентности $R = \text{matr}(r_{ij})$ графа G .

$$1. Q = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

$$2. Q = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

$$3. Q = \begin{pmatrix} 1 & 1 & 3 & 1 \\ 1 & 0 & 1 & 1 \\ 3 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$4. Q = \begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 2 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

$$5. Q = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 2 & 2 & 1 \\ 0 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

$$6. Q = \begin{pmatrix} 2 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$7. Q = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 1 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 2 & 1 & 1 & 0 \end{pmatrix}.$$

$$8. Q = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix}.$$

$$9. Q = \begin{pmatrix} 2 & 0 & 0 & 1 \\ 0 & 0 & 2 & 1 \\ 0 & 2 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

$$10. Q = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 2 & 0 & 2 \\ 1 & 0 & 1 & 3 \\ 0 & 2 & 3 & 0 \end{pmatrix}.$$

$$11. \mathbf{Q} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

$$12. \mathbf{Q} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

$$13. \mathbf{Q} = \begin{pmatrix} 1 & 2 & 1 & 1 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 3 \end{pmatrix}.$$

$$14. \mathbf{Q} = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 3 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 2 & 1 & 1 & 0 \end{pmatrix}.$$

$$15. \mathbf{Q} = \begin{pmatrix} 1 & 1 & 0 & 2 \\ 1 & 0 & 2 & 1 \\ 0 & 2 & 2 & 1 \\ 2 & 1 & 1 & 0 \end{pmatrix}.$$

$$16. \mathbf{Q} = \begin{pmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 0 & 0 & 1 & 1 \\ 1 & 1 & 1 & 2 \end{pmatrix}.$$

$$17. \mathbf{Q} = \begin{pmatrix} 2 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{pmatrix}.$$

$$18. \mathbf{Q} = \begin{pmatrix} 1 & 1 & 1 & 0 \\ 1 & 0 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 0 & 1 & 3 & 0 \end{pmatrix}.$$

$$19. \mathbf{Q} = \begin{pmatrix} 3 & 0 & 0 & 1 \\ 0 & 0 & 2 & 1 \\ 0 & 2 & 1 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

$$20. \mathbf{Q} = \begin{pmatrix} 0 & 2 & 1 & 0 \\ 2 & 2 & 0 & 2 \\ 1 & 0 & 1 & 3 \\ 0 & 2 & 3 & 0 \end{pmatrix}.$$

$$21. \mathbf{Q} = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 0 & 2 & 3 \\ 1 & 2 & 2 & 1 \\ 0 & 3 & 1 & 0 \end{pmatrix}.$$

$$22. \mathbf{Q} = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 0 & 2 & 2 & 2 \\ 1 & 2 & 0 & 1 \\ 0 & 2 & 1 & 0 \end{pmatrix}.$$

$$23. Q = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$24. Q = \begin{pmatrix} 0 & 1 & 2 & 3 \\ 1 & 0 & 1 & 1 \\ 2 & 1 & 0 & 1 \\ 3 & 1 & 1 & 0 \end{pmatrix}.$$

$$25. Q = \begin{pmatrix} 1 & 1 & 0 & 1 \\ 1 & 2 & 2 & 1 \\ 0 & 2 & 2 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

$$26. Q = \begin{pmatrix} 2 & 1 & 2 & 1 \\ 1 & 0 & 0 & 1 \\ 2 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 \end{pmatrix}.$$

$$27. Q = \begin{pmatrix} 0 & 1 & 1 & 2 \\ 1 & 0 & 2 & 0 \\ 1 & 2 & 0 & 1 \\ 2 & 0 & 1 & 0 \end{pmatrix}.$$

$$28. Q = \begin{pmatrix} 1 & 0 & 1 & 3 \\ 0 & 0 & 2 & 1 \\ 1 & 2 & 1 & 3 \\ 3 & 1 & 3 & 0 \end{pmatrix}.$$

$$29. Q = \begin{pmatrix} 2 & 0 & 3 & 1 \\ 0 & 0 & 2 & 1 \\ 3 & 2 & 0 & 0 \\ 1 & 1 & 0 & 1 \end{pmatrix}.$$

$$30. Q = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 2 & 0 & 1 \\ 1 & 0 & 1 & 3 \\ 0 & 1 & 3 & 2 \end{pmatrix}.$$

Пример решения задания 1:

Пусть дана симметричная матрица смежности неориентированного

мультиграфа $Q = \begin{pmatrix} 2 & 0 & 1 & 0 \\ 0 & 0 & 1 & 2 \\ 1 & 1 & 1 & 0 \\ 0 & 2 & 0 & 0 \end{pmatrix}.$

Данной матрице будет соответствовать граф, изображенный на рисунке 2.1.

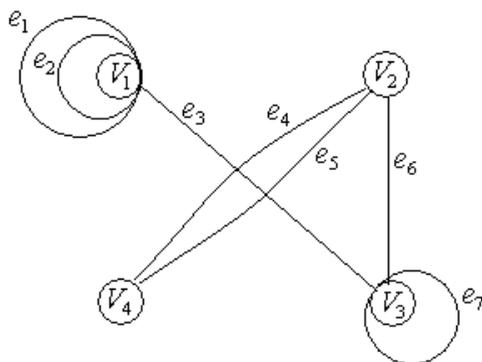
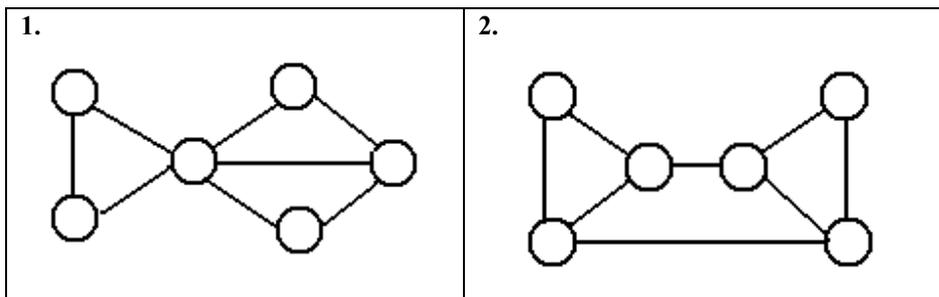


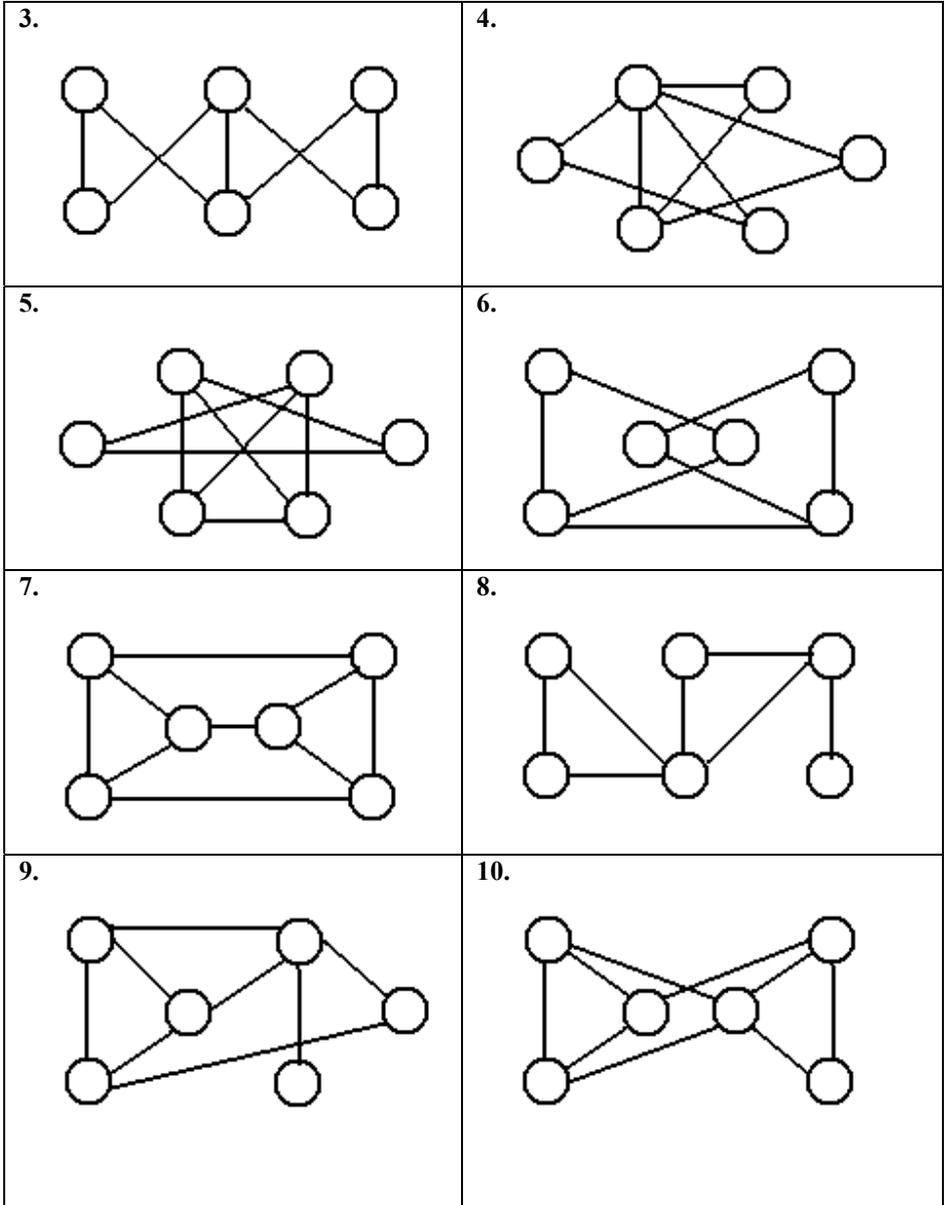
Рисунок 2.1

Если пронумеровать ребра, как указано на рисунке, то матрица инцидентности примет вид

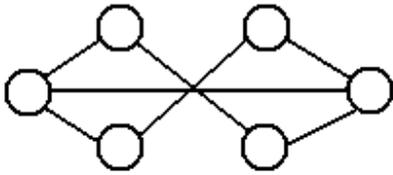
$$R = \begin{pmatrix} 2 & 2 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 2 \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}.$$

Задание 2. Дан неориентированный граф. Пронумеровать вершины. Записать матрицу смежности, раскрасить вершины, записать хроматическое число. Перерисовать, раскрасить рёбра, найти хроматический индекс. Проверить, является ли данный граф эйлеровым и гамильтоновым.

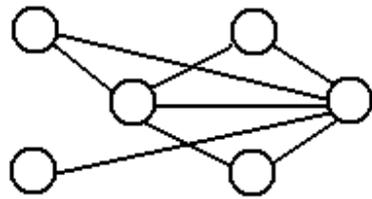




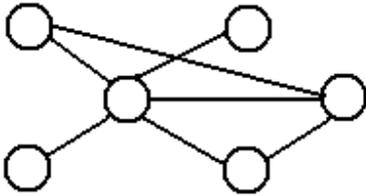
11.



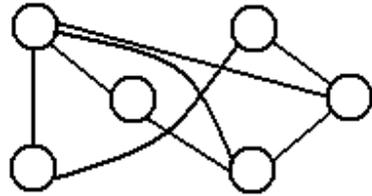
12.



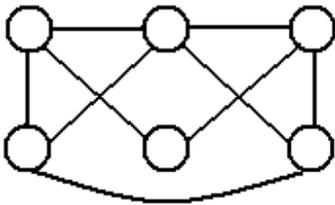
13.



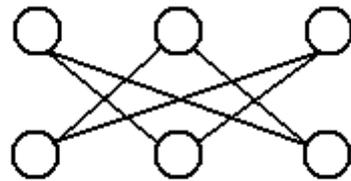
14.



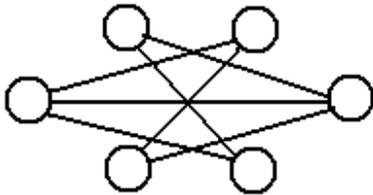
15.



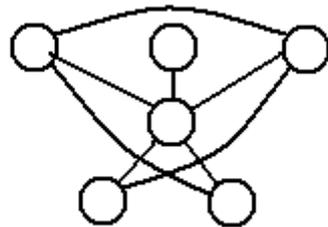
16.

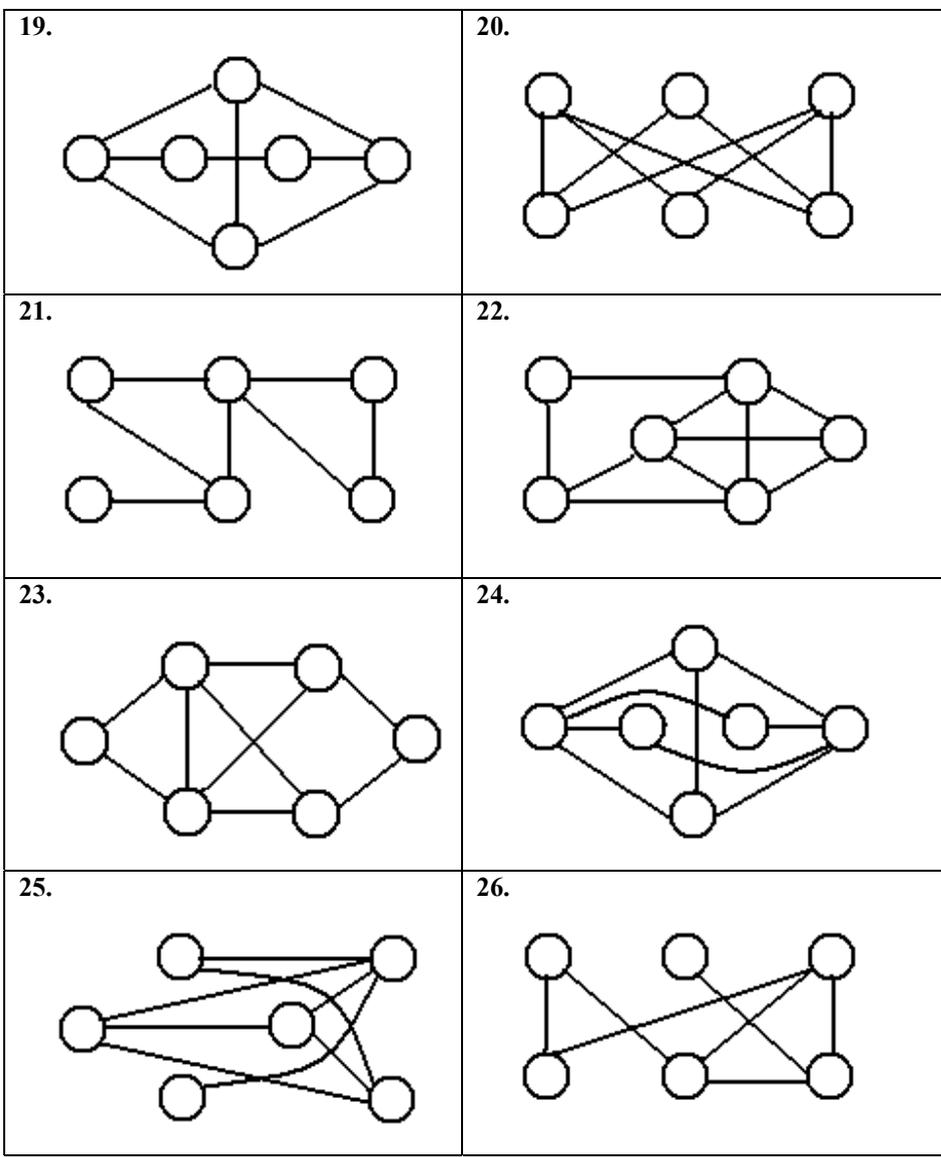


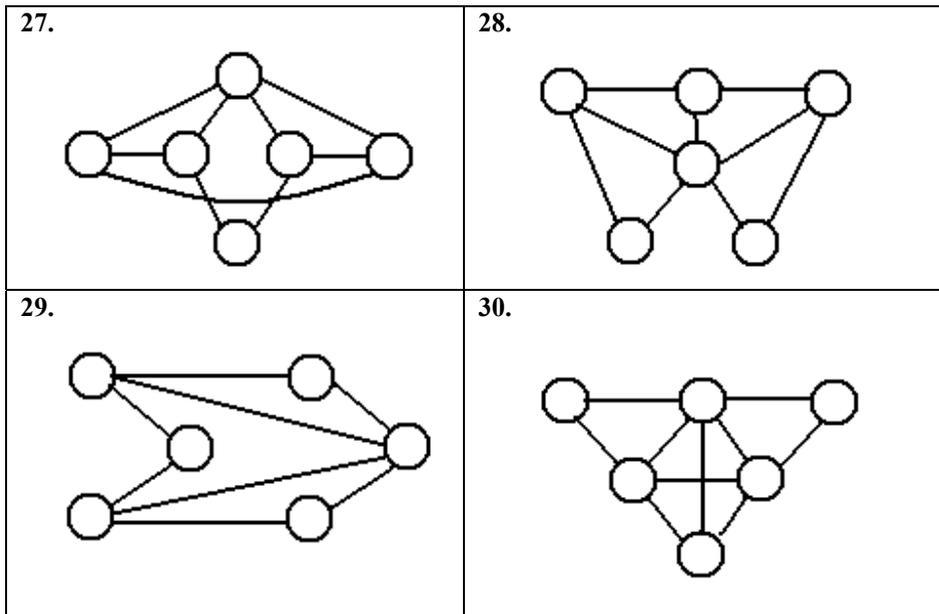
17.



18.







Пример решения задания 2:

Пусть дан следующий граф (рисунок 2.2).

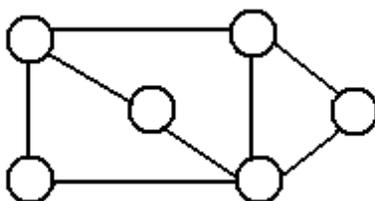


Рисунок 2.2

Пронумеруем вершины и рёбра, как указано на рисунке 2.3.

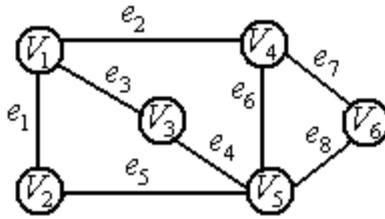


Рисунок 2.3

Запишем матрицу смежности

$$Q = \begin{pmatrix} 0 & 1 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}.$$

Матрица инцидентности примет вид

$$R = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 \end{pmatrix}.$$

Раскрасим вершины графа (рисунок 2.4). Очевидно, что хроматическое число $\chi = 3$.

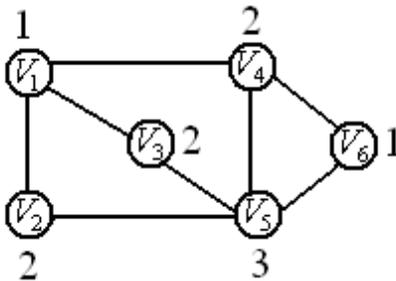


Рисунок 2.4

Раскрасим рёбра графа (рисунок 2.5). Очевидно, что хроматический индекс $g = 4$.

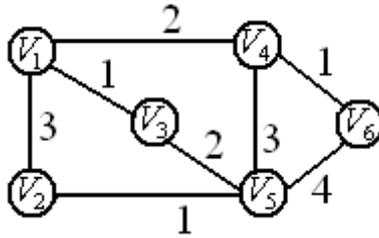


Рисунок 2.5

Проверим, является ли данный граф эйлеровым. Для этого посчитаем степени всех вершин: $d(V_1)=3$, $d(V_2)=2$, $d(V_3)=2$, $d(V_4)=3$, $d(V_5)=4$, $d(V_6)=2$. Граф содержит две вершины нечетной степени V_1 и V_4 , остальные вершины имеют чётную степень, следовательно, данный граф является полужэйлеровым. На рисунке 2.6 показан эйлеров путь.

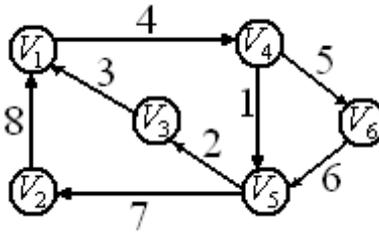


Рисунок 2.6

Данный граф является полугамильтоновым, так как содержит незамкнутый гамильтонов путь (рисунок 2.7).

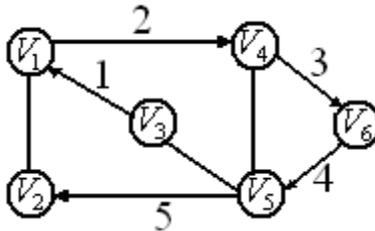


Рисунок 2.7

Задание 3. Решить задачу.

Варианты 1 – 10. Решить задачу составления расписаний.

В учебном центре необходимо провести занятия по математике, физике, химии, биологии в группах А,В,С. Занятия проводятся преподавателями К,Л,М. В табл. 2.1 – 2.10 показано, какие занятия надо провести в группах и какими преподавателями они могут быть проведены. Каждое занятие проводится в течение двух часов, включая перерывы. В центре имеются три аудитории, которые вмещают лишь одну из групп. Для занятий по физике и химии оборудована одна из этих аудиторий. Можно ли провести все необходимые занятия в течение 6 часов? Если можно, составить расписание. Если нельзя, то составить расписание, как провести эти занятия за минимальное количество часов.

Таблица 2.1. Исходные данные для задачи составления расписания.
Вариант 1

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика	+	+		+		
Физика	+	+			+	
Химия		+	+		+	
Биология	+		+			+

Таблица 2.2. Исходные данные для задачи составления расписания.
Вариант 2

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика	+		+	+		
Физика	+	+	+		+	
Химия		+			+	
Биология	+		+			+

Таблица 2.3. Исходные данные для задачи составления расписания.

Вариант 3

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика		+	+	+		
Физика	+	+			+	
Химия			+		+	
Биология	+	+	+			+

Таблица 2.4. Исходные данные для задачи составления расписания.

Вариант 4

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика		+	+	+		
Физика	+			+		
Химия	+	+	+		+	
Биология	+		+			+

Таблица 2.5. Исходные данные для задачи составления расписания.

Вариант 5

Предмет	Группа			Преподаватель	
	А	В	С	К	Л
Математика	+		+	+	
Физика	+	+		+	
Химия		+			+
Биология	+		+		+

Таблица 2.6. Исходные данные для задачи составления расписания.
Вариант 6

Предмет	Группа			Преподаватель	
	А	В	С	К	Л
Математика		+	+	+	
Физика	+		+	+	
Химия		+			+
Биология	+	+	+		+

Таблица 2.7. Исходные данные для задачи составления расписания.
Вариант 7

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика		+	+	+		
Физика	+				+	
Химия	+		+		+	
Биология	+	+	+			+

Таблица 2.8. Исходные данные для задачи составления расписания.
Вариант 8

Предмет	Группа			Преподаватель	
	А	В	С	К	Л
Математика	+	+		+	
Физика	+		+	+	
Химия		+	+		+
Биология	+				+

Таблица 2.9. Исходные данные для задачи составления расписания.

Вариант 9

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика		+	+	+		
Физика	+		+		+	
Химия		+			+	
Биология	+		+			+

Таблица 2.10. Исходные данные для задачи составления расписания.

Вариант 10

Предмет	Группа			Преподаватель	
	А	В	С	К	Л
Математика		+		+	
Физика	+		+	+	
Химия		+	+		+
Биология	+	+			+

Варианты 11 – 20. Решить задачу распределения ресурсов.

На предприятии планируется выполнить 8 работ: v_1, v_2, \dots, v_8 . Для выполнения этих работ необходимы механизмы a_1, a_2, \dots, a_6 . Использование механизмов для проведения каждой из работ определяется табл. 2.11 – 2.20. Никакой из механизмов не может быть занят одновременно на двух работах. Все работы выполняются за одно и то же время t . Как распределить механизмы, чтобы суммарное время выполнения всех работ было наименьшим?

Таблица 2.11. Исходные данные для задачи распределения ресурсов.

Вариант 11

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1	+		+				+	+
a_2		+		+				
a_3			+			+	+	
a_4	+	+			+			
a_5			+		+			+
a_6					+	+		+

Таблица 2.12. Исходные данные для задачи распределения ресурсов.

Вариант 12

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1	+		+				+	+
a_2	+			+	+			
a_3			+			+	+	
a_4	+			+	+		+	
a_5		+	+					
a_6				+		+		+

Таблица 2.13. Исходные данные для задачи распределения ресурсов.

Вариант 13

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1			+				+	+
a_2	+	+		+	+			
a_3			+			+	+	
a_4	+	+		+	+			
a_5			+				+	+
a_6					+	+		

Таблица 2.14. Исходные данные для задачи распределения ресурсов.

Вариант 14

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1		+	+				+	
a_2		+		+	+			
a_3	+		+			+		+
a_4	+	+		+				
a_5			+					+
a_6		+			+			+

Таблица 2.15. Исходные данные для задачи распределения ресурсов.

Вариант 15

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1	+		+		+		+	
a_2	+	+		+				
a_3			+			+		+
a_4	+	+		+	+			
a_5			+				+	+
a_6	+				+	+		+

Таблица 2.16. Исходные данные для задачи распределения ресурсов.

Вариант 16

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1			+				+	+
a_2	+	+		+				
a_3	+		+			+	+	
a_4	+	+		+	+			
a_5			+		+			+
a_6					+	+		+

Таблица 2.17. Исходные данные для задачи распределения ресурсов.

Вариант 17

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1		+	+			+		+
a_2	+			+		+		
a_3				+	+			
a_4	+	+			+		+	
a_5	+		+	+				
a_6		+				+		+

Таблица 2.18. Исходные данные для задачи распределения ресурсов.

Вариант 18

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1	+			+	+	+		
a_2		+	+				+	
a_3	+				+			+
a_4		+		+			+	
a_5					+	+		+
a_6	+	+	+	+				

Таблица 2.19. Исходные данные для задачи распределения ресурсов.

Вариант 19

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1		+		+		+		+
a_2			+		+			+
a_3	+					+	+	
a_4		+	+	+				
a_5						+		+
a_6			+		+	+		+

Таблица 2.20. Исходные данные для задачи распределения ресурсов.

Вариант 20

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1		+	+		+	+		
a_2		+		+			+	
a_3	+			+	+			+
a_4			+			+		
a_5	+	+	+		+		+	
a_6	+			+		+		

Варианты 21 – 30. Решить задачу экономии памяти.

На многопроцессорном вычислительном комплексе необходимо выполнить 7 заданий v_1, \dots, v_7 . Задания могут использовать общие данные (табл. 2.21 – 2.30), и в этом случае их выполнение не может проводиться одновременно. Предположим, что все задания могут выполняться за одно и то же время t . За какое наименьшее время можно выполнить все задания? Какое количество процессоров понадобится?

Таблица 2.21. Исходные данные для задачи экономии памяти.

Вариант 21

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+			+		+	+
v_2	+		+	+		+		
v_3		+		+		+		
v_4		+	+			+		
v_5	+							+
v_6		+	+	+			+	
v_7	+					+		+
v_8	+				+		+	

Таблица 2.22. Исходные данные для задачи экономии памяти.

Вариант 22

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+		+				+
v_2	+		+		+		+	+
v_3		+				+		+
v_4	+				+		+	+
v_5		+		+				+
v_6			+				+	+
v_7		+		+		+		+
v_8	+	+	+	+	+	+	+	

Таблица 2.23. Исходные данные для задачи экономии памяти.

Вариант 23

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1				+	+			+
v_2			+					
v_3		+		+	+	+	+	+
v_4	+		+					
v_5	+		+			+		+
v_6			+		+			
v_7			+					+
v_8	+		+		+		+	

Таблица 2.24. Исходные данные для задачи экономии памяти.

Вариант 24

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+		+	+	+		+
v_2	+		+	+		+	+	
v_3		+		+	+			+
v_4	+	+	+					
v_5	+		+			+	+	
v_6	+	+			+			+
v_7		+			+			+
v_8	+		+			+	+	

Таблица 2.25. Исходные данные для задачи экономии памяти.

Вариант 25

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+		+		+		+
v_2	+		+		+		+	
v_3		+		+	+			+
v_4	+		+					
v_5		+	+			+	+	
v_6	+				+			+
v_7		+			+			
v_8	+		+			+		

Таблица 2.26. Исходные данные для задачи экономии памяти.

Вариант 26

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+			+			
v_2	+		+		+		+	+
v_3		+			+		+	+
v_4					+			
v_5	+	+	+	+				+
v_6							+	
v_7		+	+			+		+
v_8		+	+		+		+	

Таблица 2.27. Исходные данные для задачи экономии памяти.

Вариант 27

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+	+	+	+			+
v_2	+		+					
v_3	+	+		+		+	+	+
v_4	+		+					
v_5	+					+	+	
v_6			+		+		+	+
v_7			+		+	+		
v_8	+		+			+		

Таблица 2.28. Исходные данные для задачи экономии памяти.

Вариант 28

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1				+			+	
v_2			+	+	+	+	+	+
v_3		+			+		+	+
v_4	+	+				+		+
v_5		+	+					
v_6		+		+			+	+
v_7	+	+	+			+		
v_8		+	+	+		+		

Таблица 2.29. Исходные данные для задачи экономии памяти.

Вариант 29

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+	+			+		+
v_2	+		+					
v_3	+	+		+	+	+		+
v_4			+		+			
v_5			+	+		+		+
v_6	+		+		+		+	+
v_7						+		
v_8	+		+		+	+		

Таблица 2.30. Исходные данные для задачи экономии памяти.

Вариант 30

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+	+	+				
v_2	+			+	+	+		+
v_3	+							
v_4	+	+				+	+	+
v_5		+						
v_6		+		+			+	+
v_7				+		+		+
v_8		+		+		+	+	

Примеры решения задания 3:

Задача составления расписаний

Решим задачу составления расписания (условие см. с. 18). В табл. 2.31 показано, какие занятия надо провести в группах и какими преподавателями они могут быть проведены.

Таблица 2.31. Исходные данные для задачи составления расписания

Предмет	Группа			Преподаватель		
	А	В	С	К	Л	М
Математика		+	+	+		
Физика	+		+	+		
Химия		+			+	
Биология	+	+	+			+

Для решения этой задачи построим граф G , вершинам которого биективно соответствуют занятия. Две вершины соединены ребром, если соответствующие занятия нельзя проводить одновременно. Ясно, что правильная раскраска графа G определяет расписание, удовлетворяющее требованиям несовместимости по времени: занятия, соответствующие вершинам, окрашенным одинаково, можно проводить одновременно. Справедливо и обратное, любое такое расписание определяет правильную раскраску графа G . Следовательно, кратчайшее время, необходимое для проведения всех занятий, равно $\chi(G) \cdot 2$, а из оптимальной раскраски графа G получается необходимое расписание.

Найдем эту раскраску. Из табл. 2.31 видно, что всего необходимо провести 8 занятий. Обозначим занятия в каждой группе заглавными буквами (М,В) – математика в группе В и т.д. Нарисуем граф, где рёбрами соединим вершины занятий, которые нельзя проводить одновременно. Например (М,В), (М,С), (Ф,А), (Ф,С) нельзя проводить одновременно, так как их проводит один преподаватель. Кроме того, соединим рёбрами все занятия, проводимые в одной группе, далее занятия по физике и химии, так как для них оборудована только одна из этих аудиторий. В итоге получим следующий граф (рис. 2.8).

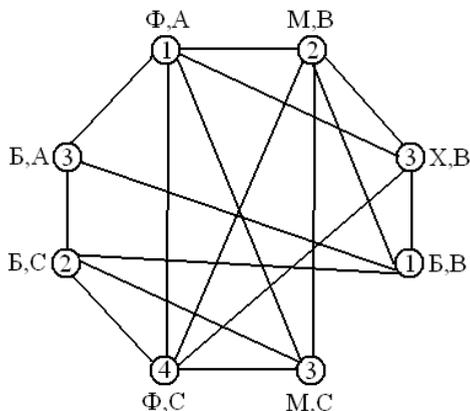


Рисунок 2.8

Так как данный граф содержит связный четырехвершинный подграф $((M,B), (M,C), (F,A), (F,C))$, то для раскрашивания вершин потребуется не менее 4 цветов. На рисунке 2.8 показана раскраска данного графа. Хроматическое число равно 4. В итоге получили, что занятия нельзя провести за 6 часов, только за 8. Запишем расписание, как это сделать. Получим следующую таблицу (таблица 2.32). В неё выписаны по строкам занятия, соответствующие цветам.

Таблица 2.32. Предварительное расписание занятий

Группа	А	В	С
1 пара	Физика	Биология	
2 пара		Математика	Биология
3 пара	Биология	Химия	Математика
4 пара			Физика

В данном расписании у группы А получилось «окно» на второй паре. Для того, чтобы его убрать поменяем 2ю и 3ю строки местами. Получили окончательное расписание (таблица 2.33)

Таблица 2.33. Окончательное расписание занятий

Группа	А	В	С
1 пара	Физика	Биология	
2 пара	Биология	Химия	Математика
3 пара		Математика	Биология
4 пара			Физика

Задача распределения ресурсов

Решим задачу распределения ресурсов (условие см. с. 21).
Использование механизмов для проведения каждой из работ определяется таблицей 2.34.

Таблица 2.34. Исходные данные для задачи распределения ресурсов

Работа механ.	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
a_1		+	+		+	+		
a_2		+		+			+	
a_3	+			+	+			+
a_4			+			+		
a_5	+	+			+		+	
a_6	+			+		+		

Рассмотрим граф G с множеством вершин V . Две различные вершины v_i и v_j графа G смежны тогда и только тогда, когда для выполнения работ v_i и v_j требуется хотя бы один общий механизм. Наименьшее время выполнения всех работ равно $\chi(G) \cdot t$. Оптимальная раскраска графа G определяет распределение механизмов.

Нарисуем граф, соответствующий данной таблице (см. рисунок 2.9).

Так как данный граф содержит связные четырёхвершинные подграфы, то для раскрашивания вершин потребуется на менее 4 цветов. Для раскрашивания вершин данного графа потребовалось 5 цветов. Следовательно, хроматическое число равно 5 и время выполнения работ $5 \cdot t$. Работы можно выполнять в следующем порядке:

1. V_1 V_3
2. V_2 V_8
3. V_5
4. V_6 V_7
5. V_4

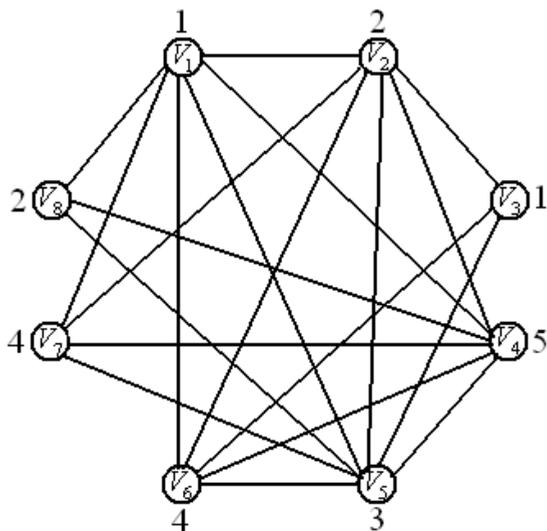


Рисунок 2.9

Задача экономии памяти

Решим задачу экономии памяти (условие см. с. 25). Использование общих данных при выполнении заданий отражено в таблице 2.35.

Таблица 2.35. Исходные данные для задачи экономии памяти

Задания	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1		+		+	+	+	+	
v_2	+		+			+		+
v_3		+					+	+
v_4	+				+	+		
v_5	+			+			+	+
v_6	+	+		+				+
v_7	+		+		+			+
v_8		+	+		+	+	+	

На множестве заданий $V = \{ v_1, \dots, v_8 \}$ введем структуру графа: два задания соединим ребром, если они используют общие данные. Полученный граф будем называть графом несовместимости заданий. Задания не могут выполняться одновременно, если используют общие данные. Следовательно, задача сводится к нахождению оптимальной раскраски графа несовместимости. Наименьшее время выполнения всех заданий равно $\chi(G) \cdot t$. Количество процессоров равно максимальному числу вершин, раскрашенных одним цветом.

Нарисуем граф, соответствующий данной таблице (см. рисунок 2.10). Для раскрашивания вершин данного графа потребовалось 3 цвета. Следовательно, хроматическое число равно 3 и время выполнения заданий $3 \cdot t$.

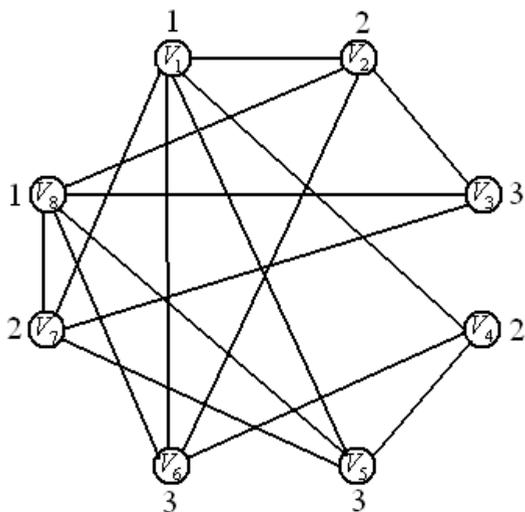


Рисунок 2.10

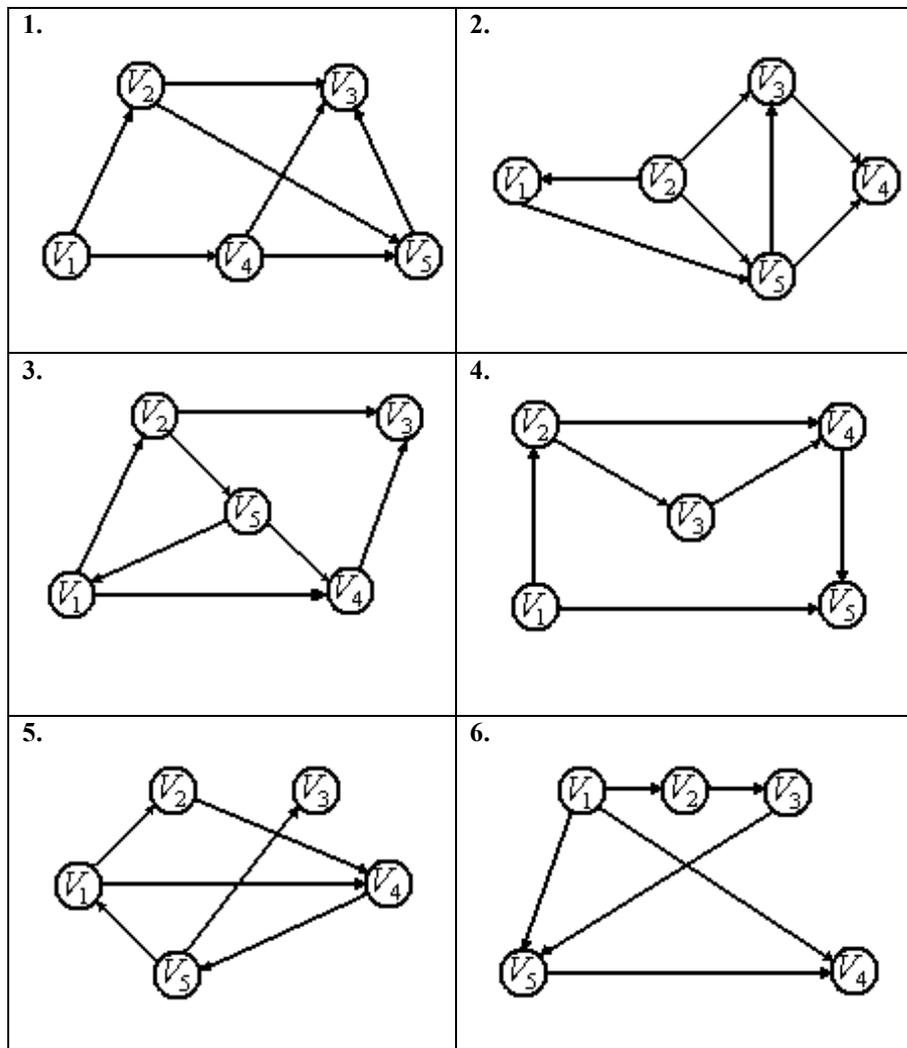
Задания можно выполнять в следующем порядке:

1. $V_1 \ V_8$
2. $V_2 \ V_4 \ V_7$
3. $V_3 \ V_5 \ V_6$

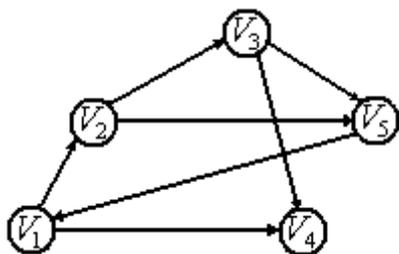
Потребуется три процессора.

3. ОРИЕНТИРОВАННЫЕ ГРАФЫ

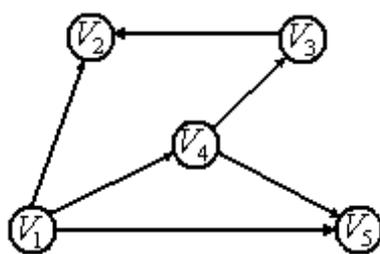
Задание 1. Дан ориентированный граф. Записать матрицу смежности, матрицу инцидентности. Найти кратчайшие расстояния от первой вершины. Проверить, является ли он эйлеровым и гамильтоновым.



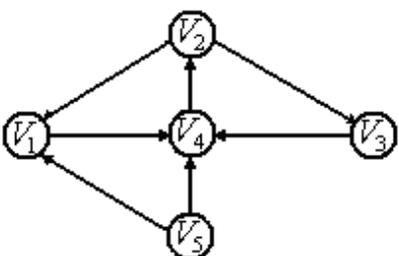
7.



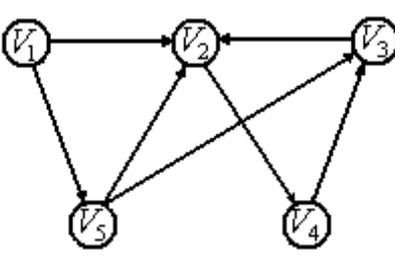
8.



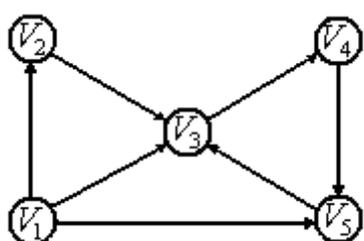
9.



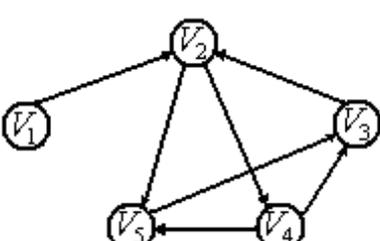
10.



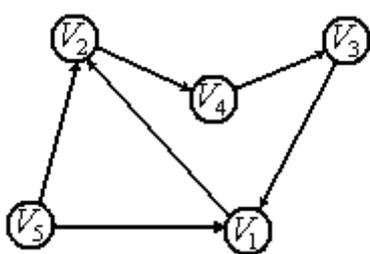
11.



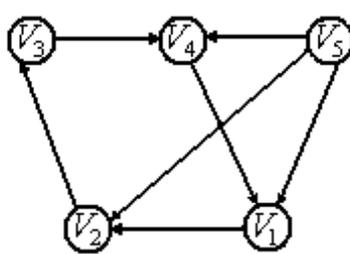
12.

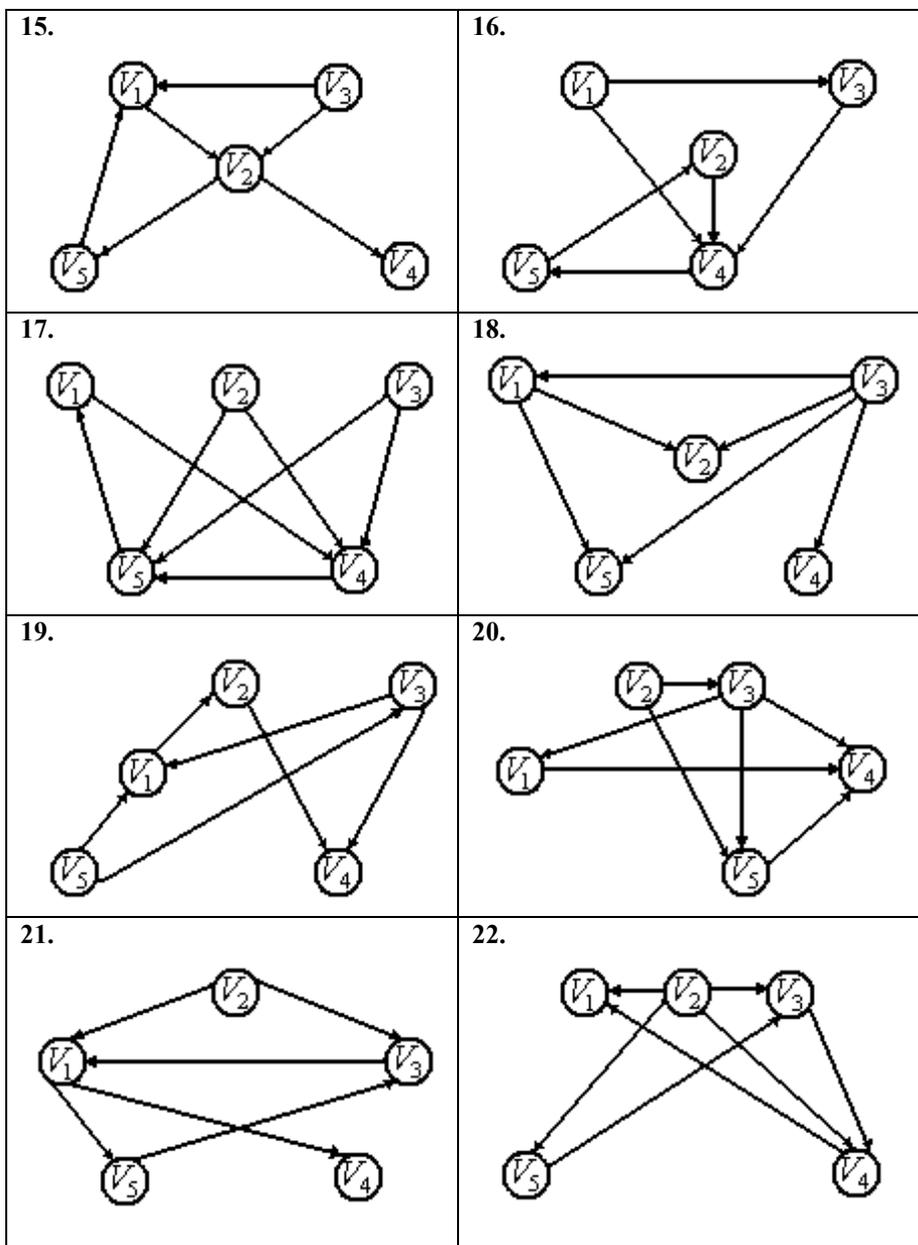


13.



14.





Пример решения задания 1:
 Пусть дан следующий граф (рисунок 3.1)

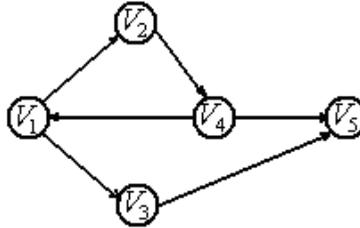


Рисунок 3.1

Пронумеруем рёбра, как указано на рисунке 3.2

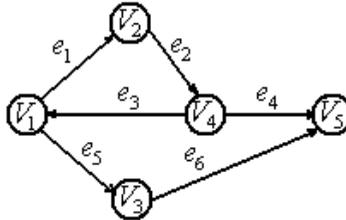


Рисунок 3.2

Запишем матрицу смежности

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

Матрица инцидентности примет вид

$$R = \begin{pmatrix} +1 & 0 & -1 & 0 & +1 & 0 \\ -1 & +1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & +1 \\ 0 & -1 & +1 & +1 & 0 & 0 \\ 0 & 0 & 0 & -1 & 0 & -1 \end{pmatrix}.$$

Найдём кратчайшие расстояния от первой вершины. Так как граф не взвешенный, то расстояние равно числу дуг, соединяющих вершины. Получаем $d(V_1, V_2)=1$, $d(V_1, V_3)=1$, $d(V_1, V_4)=2$, $d(V_1, V_5)=2$.

Проверим, является ли данный граф эйлеровым. Данный граф не является эйлеровым, так как он не является сильносвязанным, например из вершины V_5 нельзя попасть ни в одну вершину. Кроме того, если посчитать полустепени исхода и захода всех вершин: $\rho^-(V_1)=2, \rho^+(V_1)=1, \rho^-(V_2)=1, \rho^+(V_2)=1, \rho^-(V_3)=1, \rho^+(V_3)=1, \rho^-(V_4)=2, \rho^+(V_4)=1, \rho^-(V_5)=0, \rho^+(V_5)=2$, увидим, что у вершин V_1, V_4, V_5 полустепени исхода и захода не равны.

Также он не является гамильтоновым, потому что из вершины V_5 не выходит ни одного ребра.

Задание 2. Решить задачу

Варианты 1 - 10.

Дана структурно-временная таблица перечня работ по организации презентации товаров (таблица 3.1). Постройте сетевой график, определите критический путь, критические работы, резерв времени.

Таблица 3.1. Структурно-временная таблица перечня работ по организации презентации товаров

Номер и содержание работы	Предшествующие работы	Длительность работы по вариантам									
		1	2	3	4	5	6	7	8	9	10
1. Заказ на оборудование и товары	-	12	9	14	10	8	11	13	12	8	10
2. Разработка системы учёта товаров и оборудования	-	11	10	12	13	15	11	14	9	10	8
3. Завоз товаров	a_1	10	9	8	7	6	5	4	3	2	10
4. Завоз оборудования	a_1	3	4	5	6	7	6	5	4	7	5
5. Установка оборудования	a_4	6	7	8	9	10	5	6	8	3	7
6. Учёт товаров	a_2, a_3	2	3	5	4	3	6	2	4	3	5
7. Выкладка товаров	a_3, a_5	2	3	2	4	6	3	5	2	5	4
8. Оформление зала	a_7	6	6	5	6	4	6	5	7	5	8
9. Презентация	a_8	2	4	2	3	2	3	5	2	3	6
10. Анализ результатов	a_6, a_9	3	2	4	2	2	2	4	2	3	3

Варианты 11 - 20.

Дана структурно-временная таблица перечня работ по подготовке к проведению дополнительного обучения (таблица 3.2). Постройте сетевой график, определите критический путь, критические работы, резерв времени.

Таблица 3.2. Структурно-временная таблица перечня работ по подготовке к проведению дополнительного обучения

Номер и содержание работы	Предшествующие работы	Длительность работы по вариантам									
		11	12	13	14	15	16	17	18	19	20
1. Разработка учебного плана	-	9	9	14	10	8	11	13	12	8	10
2. Подготовка учебных материалов	-	25	27	12	40	35	31	24	29	30	45
3. Заключение договора	a_1	6	7	8	7	6	5	4	3	2	6
4. Согласование учебного плана	a_1	3	4	5	6	7	6	5	4	7	5
5. Подготовка списка обучающихся	a_3, a_4	6	7	8	9	10	5	6	8	3	7
6. Утверждение учебного плана	a_4	2	3	5	4	3	6	2	4	3	5
7. Поиск и согласование аудиторий	a_3	4	3	5	4	6	4	5	6	5	4
8. Составление расписания	a_5, a_6, a_7	5	6	5	6	4	6	4	6	5	8
9. Издание приказа	a_5, a_6	2	4	2	3	2	3	5	2	3	6
10. Комплектование учебных материалов	a_2	3	2	3	2	2	2	4	2	2	3

Варианты 21 - 30.

Дана структурно-временная таблица перечня работ по подготовке к проведению конференции (таблица 3.3). Постройте сетевой график, определите критический путь, критические работы, резерв времени.

Таблица 3.3. Структурно-временная таблица перечня работ по подготовке к проведению конференции

Номер и содержание работы	Предшествующие работы	Длительность работы по вариантам									
		21	22	23	24	25	26	27	28	29	30
1. Утвердить дату проведения	–	3	2	1	2	2	2	1	2	1	3
2. Проверить состояние возможных помещений	–	4	3	5	4	3	6	2	4	3	5
3. Составить повестку дня	–	5	4	4	7	6	5	4	3	5	3
4. Выбрать помещение	a_1, a_2	3	4	5	6	4	6	5	4	4	5
5. Составить и напечатать приглашения	a_3, a_4	2	3	2	4	3	3	2	3	3	4
6. Разослать приглашения	a_5	2	3	3	4	3	2	2	4	3	5
7. Собрать тексты докладов	a_3	9	9	14	10	8	11	13	12	8	10
8. Напечатать сборник докладов	a_7	8	6	5	6	10	6	7	7	9	8
9. Организовать питание	a_4	2	4	2	3	2	3	5	2	3	6
10. Оформить помещение	a_4	3	2	1	2	2	2	1	2	1	3

Пример решения задания 2:

Рассмотрим задачу о выполнении некоторой совокупности работ. Для каждой работы известно время выполнения и непосредственно предшествующие работы, т.е. те работы, без выполнения которых данная работа выполняться не может. Такая совокупность работ называется проектом. Пусть дана следующая структурно-временная таблица перечня работ, состоящая из семи работ (таблица 3.4).

Таблица 3.4. Структурно-временная таблица перечня работ

Наименование работы	Время выполнения	Предшествующие работы
v_1	3	--
v_2	3	--
v_3	2	v_1
v_4	4	v_1, v_2
v_5	2	v_2
v_6	5	v_3, v_4
v_7	3	v_5

Для решения задачи представим проект в виде ориентированного графа, вершинами которого являются работы, а дуги вводятся следующим образом: если работа v_i непосредственно предшествует работе v_j , то изображается (v_i, v_j) -дуга. Над такой дугой проставляется время выполнения работы v_j . Полученная сеть не содержит контуров, так как работы должны выполняться последовательно. Следовательно, сеть имеет хотя бы один источник и хотя бы один сток.

Для решения задачи удобно расширить проект, введя две «фиктивные» работы: «начало проекта» и «завершение проекта». «Начало проекта» предшествует всем источникам. Все стоки предшествуют работе «завершение проекта». Время выполнения этих работ равно нулю. Полученная расширенная сеть называется *сетевым графиком проекта*. Сетевой график проекта изображен на рисунке 3.3. В данном проекте фиктивные работы – v_0 и v_8 .

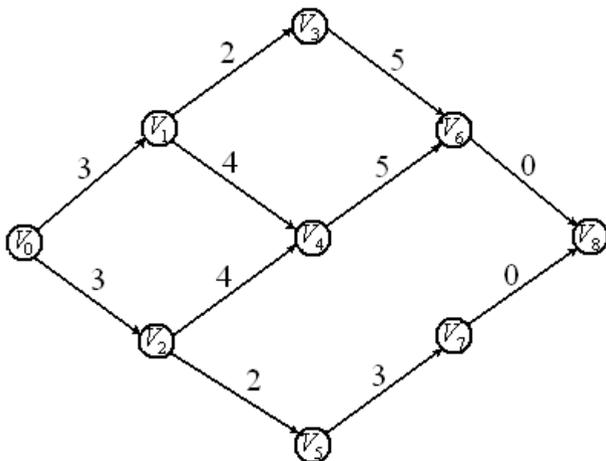


Рисунок 3.3

Будем считать, что вершины сетевого графика занумерованы так, что если v_i предшествует v_j , то $i < j$. Если это не так, следует перенумеровать работы проекта, чтобы это условие выполнялось.

Сначала определим наименьшее время, необходимое для выполнения всего проекта. Введем ряд обозначений: $t(v_i)$ – время выполнения работы v_i ; T_{\min} – наименьшее время выполнения всего проекта; $PВН(v_i)$ и $PВО(v_i)$ – наиболее раннее время начала и, соответственно, наиболее раннее время окончания работы v_i ; v_0 – источник сетевого графика; v_k – сток (в нашем примере v_8). Таким образом, $T_{\min} = PВН(v_k) = PВО(v_k)$.

Полагаем, что $PВН(v_0) = PВО(v_0) = 0$. Для любой вершины $v_i \neq v_0$ сетевого графика справедливы равенства $PВН(v_i) = \max \{ PВО(v') \mid v' \text{ предшествует } v_i \}$, $PВО(v_i) = PВН(v_i) + t(v_i)$.

Вычислим таким образом величины $PВН(v)$ и $PВO(v)$ для всех вершин v сетевого графика. Пусть далее указанные величины найдены для всех вершин v_0, v_1, \dots, v_{l-1} . Тогда поскольку все вершины, предшествующие v_l , находятся среди v_0, v_1, \dots, v_{l-1} , то полагаем $PВН(v_l) = \max\{PВO(v') \mid v' \text{ предшествует } v_l\}$ и $PВO(v_l) = PВН(v_l) + t(v_l)$.

Например, для данного графика $PВН(v_1) = 0$, $PВO(v_1) = 3$; $PВН(v_2) = 0$, $PВO(v_2) = 3$; $PВН(v_3) = 3$ (так как этой работе предшествует работа v_1), $PВO(v_3) = 5$; $PВН(v_4) = 3$ (этой работе предшествуют работы v_1 и v_2), $PВO(v_4) = 7$; $PВН(v_5) = 3$ (этой работе предшествует работа v_2), $PВO(v_5) = 5$, $PВН(v_6) = 7$ (этой работе предшествуют работы v_3 и v_4 и не можем начать её выполнение раньше времени окончания работы v_4), $PВO(v_6) = 12$; $PВН(v_7) = 5$, $PВO(v_7) = 8$; $PВН(v_8) = 12$, $PВO(v_8) = 12$ – завершение проекта. Результат вычисления для данного сетевого графика приведен в таблице 3.5. Таким образом, получили $T_{\min} = 12$.

Наиболее раннее время окончания работы v_j равно длине самого длинного пути из v_0 в v_j . Заметим, что самый длинный путь из v_0 в v_j может быть не один.

Теперь определим критические работы. Предположим, что уже вычислено наиболее раннее время окончания каждой работы и всего проекта в целом. Тогда возникает вопрос о нахождении самого позднего времени окончания работ, которое не влияет на время выполнения всего проекта в целом, то есть нахождении запаса времени для каждой работы.

Введём соответствующие обозначения: $PВН(v_i)$ и $PВO(v_i)$ – соответственно наиболее позднее время начала и наиболее позднее время окончания работы v_i , не ведущие к увеличению времени выполнения проекта. Работа v_i , для которой $PВO(v_i) = PВН(v_i)$, называется *критической*. Временному графику выполнения критических работ необходимо уделять особое внимание, поскольку сдвиг во времени при выполнении этих работ повлечёт увеличение времени выполнения всего проекта в целом. Для любой вершины $v_i \neq v_k$ сетевого графика справедливы равенства $PВO(v_i) = \min\{PВН(v') \mid v' \text{ предшествует } v_i\}$, $PВН(v') = PВO(v') - t(v')$.

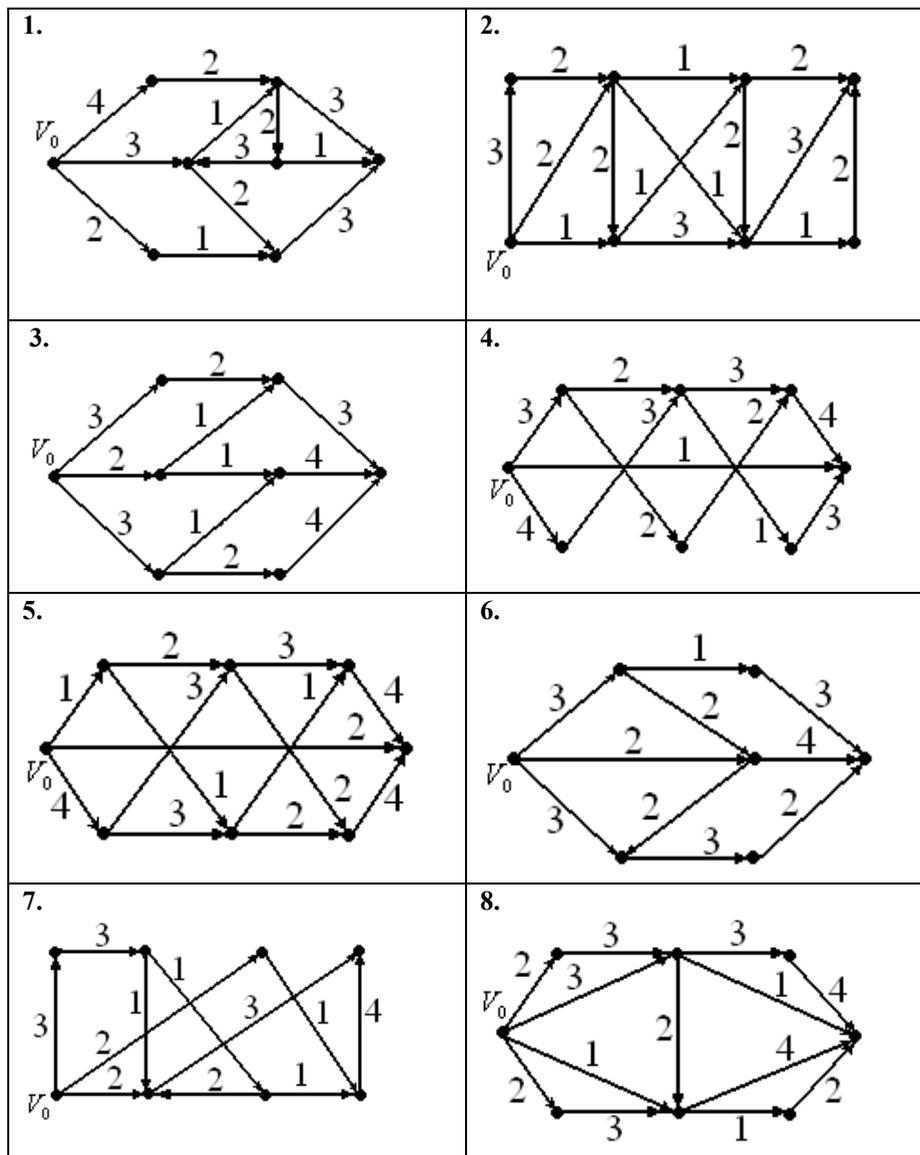
Например, для данного графика $PВO(v_8) = 12$, $PВН(v_8) = 12$, работа v_8 является критической; $PВO(v_7) = 12$ и $PВO(v_6) = 12$ (так как v_6 и v_7 являются предшествующими для работы v_8) у работы v_6 , кроме того, позднее время

окончания совпадает с ранним временем окончания, следовательно она является критической работой; $ПВН(v_7)=9$, $ПВН(v_6)=7$; аналогично получаем $ПВО(v_5)=9$, $ПВН(v_5)=7$; $ПВО(v_4)=7$; $ПВН(v_4)=3$; $РВО(v_3)=7$, $РВН(v_3)=5$; $РВО(v_2)=3$ (работа v_2 не может быть завершена позднее, чем позднее время начала работ v_4, v_3), $РВН(v_2)=0$; $РВО(v_1)=3$, $РВН(v_1)=0$. Результат вычисления величин $ПВН(v)$ и $ПВО(v)$ для сетевого графика, изображенного на рисунке 3.3, в случае, когда $T=T_{\min}=12$, приведен в таблице 3.5. Как мы видим, критическими работами в этом примере являются работы $v_0, v_1, v_2, v_4, v_6, v_8$. Запас времени для остальных работ вычислен по формуле $\Delta(v_i)=ПВО(v_i) - РВО(v_i)$. Любая работа v_i не может быть начата раньше $РВН(v_i)$ и окончена позднее $ПВО(v_i)$.

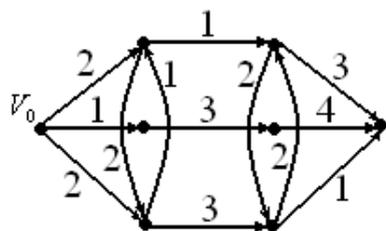
Таблица 3.5. Результаты вычисления времени начала и окончания работ

Работы	$РВН(v_i)$	$РВО(v_i)$	$ПВН(v_i)$	$ПВО(v_i)$	Запас времени
v_0	0	0	0	0	кр
v_1	0	3	0	3	кр
v_2	0	3	0	3	кр
v_3	3	5	5	7	2
v_4	3	7	3	7	кр
v_5	3	5	7	9	4
v_6	7	12	7	12	кр
v_7	5	8	9	12	4
v_8	12	12	12	12	кр

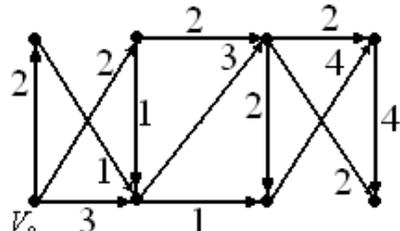
Задание 3. Дан ориентированный взвешенный граф. С помощью алгоритма Дейкстры вычислить расстояния от нулевой вершины до остальных.



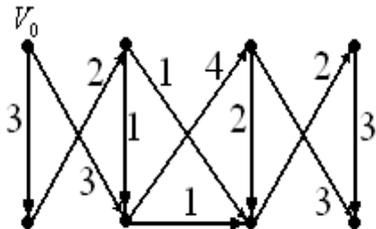
9.



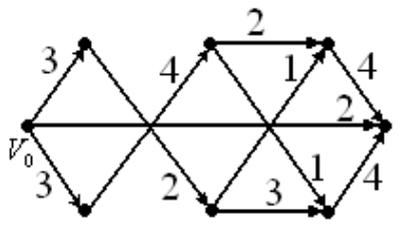
10.



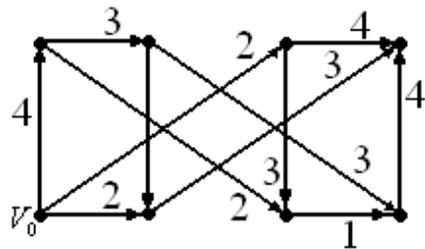
11.



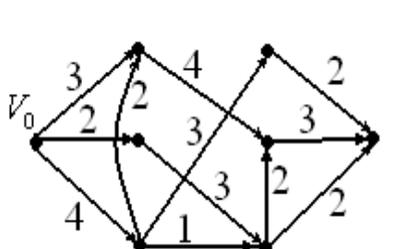
12.



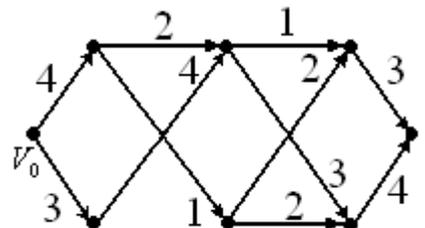
13.



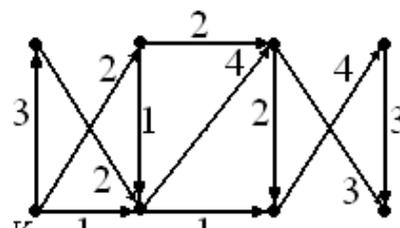
14.

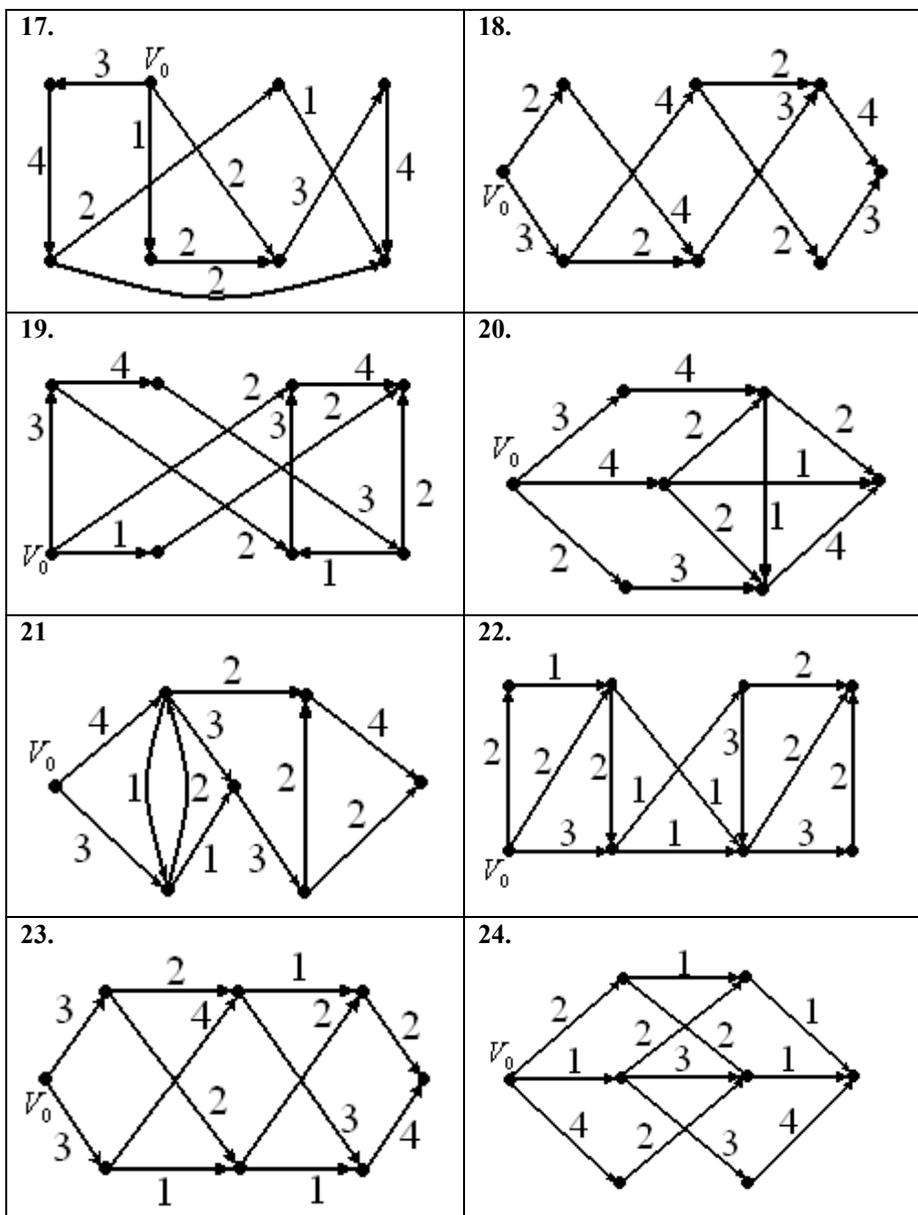


15.

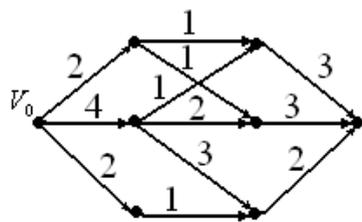


16.

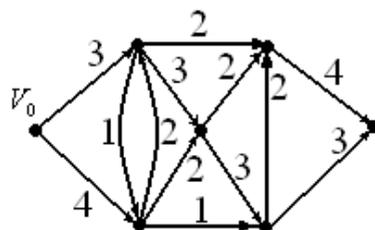




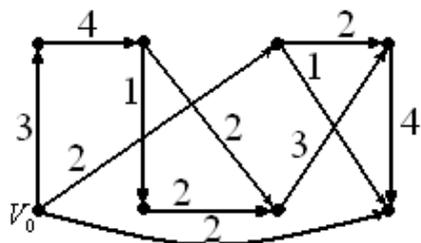
25.



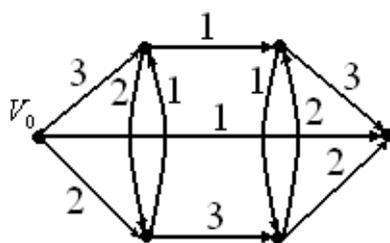
26.



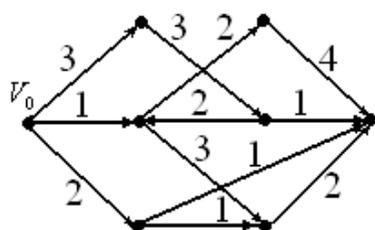
27.



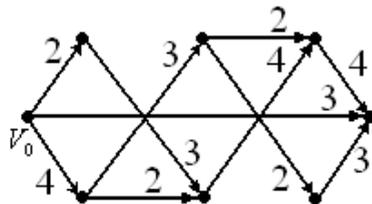
28.



29.



30.



Пример решения задания 3:

Алгоритм Дейкстры вычисляет расстояние от фиксированной вершины v_0 до всех остальных вершин сети. Прежде чем привести описание алгоритма, введём необходимые обозначения. Пусть $N=(G, \alpha)$ – сеть, v_0 – фиксированная вершина этой сети, v_1, \dots, v_k – остальные вершины. Для $i=1, \dots, k$ расстояние $d(v_0, v_i)$ обозначим через $d(v_i)$. На множестве вершин введем двухместную функцию $a(v_i, v_i')$ следующим образом:

$$a(v_i, v_i') = \begin{cases} \alpha(e), & \text{если } e \text{ – дуга с началом } v_i \text{ и концом в } v_i', \\ \infty, & \text{если не существует дуги } (v_i, v_i'). \end{cases}$$

Описание алгоритма, кроме того, еще использует подмножество S множества вершин V .

Алгоритм Дейкстры:

Шаг 1. Полагаем $S = V \setminus \{v_0\}$, $d(v_i) = a(v_0, v_i)$ для всех $i=1, \dots, k$.

Шаг 2. Если $|S|=1$, то выполнить шаг 4. Иначе выбрать в S элемент v' , для которого величина $d(v')$ является наименьшей (среди всех элементов из S).

Шаг 3. Положить $S = S \setminus \{v'\}$, $d(v) = \min\{d(v), d(v') + a(v', v)\}$ для всех $v \in S$ и перейти к шагу 2.

Шаг 4. Выдать $d(v_1), \dots, d(v_k)$. Конец работы алгоритма.

Алгоритм осуществляет «итеративный» процесс нахождения расстояния $d(v_i)$ для $i=1, \dots, k$. Проследим, как это делается для сети N , изображенной на рисунке 3.4. Работа алгоритма будет иллюстрироваться заполнением таблицы 3.6.

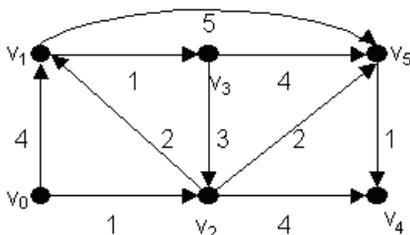


Рисунок 3.4

При выполнении первого шага алгоритма величины $d(v_i)$ примут значения, которые указаны в нулевой строке таблицы 3.6. То есть $d(v_1)=4$, $d(v_2)=1$, $d(v_3)=\infty$, $d(v_4)=\infty$, $d(v_5)=\infty$ (три последних расстояния приняли такие значения, так как не существует дуг из вершины v_0 в эти вершины). Из этих значений точным является только $d(v_2)$, значения остальных величин завышены. На шаге 2 в качестве v' выбирается вершина v_2 (в таблице соответствующее значение набрано жирным шрифтом) и удаляется из S на шаге 3. На том же шаге уточняются предыдущие значения $d(v)$ для $v \in S$ (см. первую строку таблицы 3.6). Получаем $d(v_1)=d(v_2)+a(v_2, v_1)=1+2=3$, $d(v_3)=\infty$ (так как дуги (v_2, v_3) не существует), $d(v_4)=d(v_2)+a(v_2, v_4)=1+4=5$, $d(v_5)=d(v_2)+a(v_2, v_5)=1+2=3$. На этом первый проход алгоритма заканчивается. На втором проходе в качестве v' берется v_5 , удаляется из S и значения $d(v)$ для $v \in S$ снова уточняются. Значения $d(v_1)$, $d(v_3)$ остаются прежними, так как не существует дуг из вершины v_5 в эти вершины, $d(v_4)=d(v_5)+a(v_5, v_4)=3+1=4$, полученное значение меньше предыдущего, поэтому записываем его в таблицу. Результат отражен во второй строке таблицы 3.6. На третьем проходе в качестве v' берется v_1 . Получаем $d(v_3)=d(v_1)+a(v_1, v_3)=3+1=4$, $d(v_4)$ остаётся прежним, полученный результат записываем в третью строку таблицы 3.6. На четвертом проходе в качестве v' берём v_4 . Значение $d(v_3)$ остаётся прежним. Теперь, когда множество S содержит только вершину v_3 , алгоритм заканчивает работу. Результат: $d(v_1)=3$, $d(v_2)=1$, $d(v_3)=4$, $d(v_4)=4$, $d(v_5)=3$.

Таблица 3.6. Пример заполнения таблицы для алгоритма Дейкстры

Номер строки	$d(v_1)$	$d(v_2)$	$d(v_3)$	$d(v_4)$	$d(v_5)$
0	4	1	∞	∞	∞
1	3		∞	5	3
2	3		∞	4	
3			4	4	
4			4		

Задание 4. Для сети из задания 3 найти максимальный поток.

Пример решения задания 4:

Алгоритм нахождения максимального потока

Дана сеть (G, α) , a – источник, b – сток сети.

Шаг 1. Если не существует пути из источника в сток, то положить $\varphi=0$ и перейти к шагу 4, иначе выбрать непустое множество T

непересекающихся по дугам путей из a в b . Если e_1, e_2, \dots, e_k – путь из T , т.е. последовательность дуг, то положить $\varphi(e_1)=\varphi(e_2)=\dots=\varphi(e_k)=\min\{\alpha(e_1), \dots, \alpha(e_k)\}$. Для дуг e , через которые не проходят пути из T , положить $\varphi(e)=0$. В результате получаем ненулевой поток φ через сеть (G, a) .

Шаг 2. Исходя из сети (G, a) и потока φ построить сеть (G', a') следующим образом. Граф G' будет иметь те же вершины, что и граф G . Если e – дуга графа G и $\alpha(e) - \varphi(e) \neq 0$, то e – дуга графа G' и $\alpha'(e) = \alpha(e) - \varphi(e)$. Если e – дуга графа G и $\varphi(e) \neq 0$, то вводим дугу \bar{e} обратной ориентации, нежели e , и полагаем $\alpha'(\bar{e}) = \varphi(e)$. В случае, когда возникают кратные дуги e_1 и e_2 , вводим вместо них одну дугу e и полагаем $\alpha'(e) = \alpha'(e_1) + \alpha'(e_2)$.

Шаг 3. Если в сети (G', a') не существует пути из a в b , то перейти к шагу 4, иначе в сети (G', a') построить ненулевой поток φ' так, как это предписано шагом 1. Для сети (G, a) положить $\varphi = \varphi + \varphi'$ и перейти к шагу 2.

Шаг 4. Выдать φ . Конец работы алгоритма.

На примере сети, изображенной на рисунке 3.5, проиллюстрируем работу алгоритма.

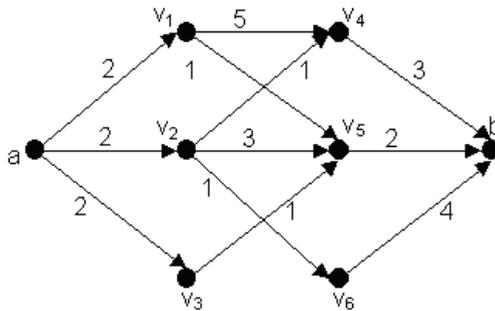


Рисунок 3.5

В качестве множества T на первом шаге алгоритма выбрано множество из двух путей: 1-й: a, v_1, v_4, b и 2-й: a, v_2, v_5, b (рисунок 3.6). Пропускная способность первого пути: $\varphi_1 = \varphi(a, v_1) = \varphi(v_1, v_4) = \varphi(v_4, b) = 2$. Пропускная способность второго пути: $\varphi_2 = \varphi(a, v_2) = \varphi(v_2, v_5) = \varphi(v_5, b) = 2$. Для остальных дуг $\varphi(e) = 0$ (поток указан в скобках). Суммарный поток через эту сеть равен $\varphi = \varphi_1 + \varphi_2 = 2 + 2 = 4$.

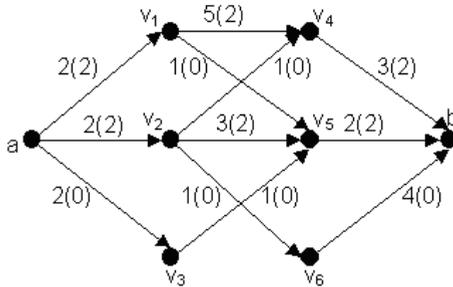


Рисунок 3.6

Приступим к шагу 2. Построим новый граф (рисунок 3.7), который будет иметь те же вершины, что и исходный. Для дуг из первых двух путей построим обратные дуги, пропускная способность которых будет равна потоку, пропущенному через них на первом шаге. Также построим дуги исходного графа, пропускная способность которых получается вычитанием из исходной пропускной способности потока, пропущенного на первом шаге. Дуги с нулевой пропускной способностью строить не будем. Так вершины v_1, v_4 будут соединены дугами (v_1, v_4) с пропускной способностью $5-2=3$ и дугой (v_4, v_1) с пропускной способностью 2. Вершины v_1, a будут соединены только дугой (v_1, a) с пропускной способностью 2.

На третьем шаге в качестве множества Γ для построения потока φ' возьмём множество, состоящее из одного пути: a, v_3, v_5, v_2, v_6, b . Пропускная способность этого пути: $\varphi' = \varphi(a, v_3) = \varphi(v_3, v_5) = \varphi(v_5, v_2) = \varphi(v_2, v_6) = \varphi(v_6, b) = 1$. На рисунке 3.7 представлена сеть (G', α') , полученная после выполнения шага 2 из сети, изображенной на рисунке 3.5, с указанным (в скобках) потоком φ' .

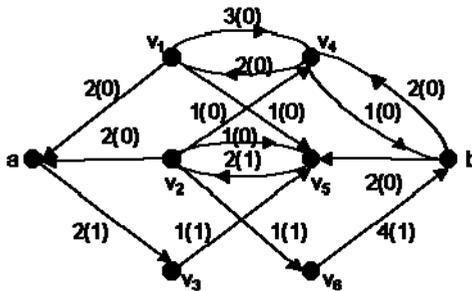


Рисунок 3.7

Поток $\varphi+\varphi'$ для сети (G, α) изображен на рисунке 3.8. Этим завершен один проход алгоритма. Обратим внимание на то, что для $e=(v_2, v_5)$ $\varphi+\varphi'(e)=\varphi(e)-\varphi'(\bar{e})=1$. Суммарный поток $\varphi=\varphi+\varphi'=4+1=5$.

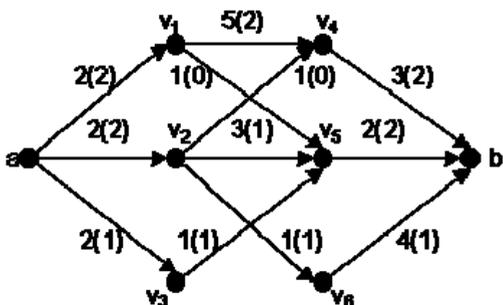


Рисунок 3.8

Так закончен третий шаг первого прохода алгоритма. Теперь следует перейти ко второму проходу алгоритма. Сеть, построенная на шаге 2 второго прохода алгоритма, уже не имеет (a, b) – путей (рисунок 3.9).

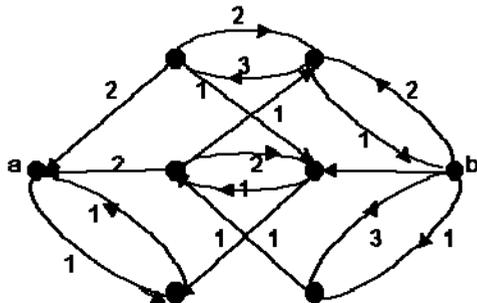


Рисунок 3.9

Следовательно, мы должны перейти к четвертому шагу и выдать φ . Конец работы алгоритма. На рисунке 3.8 изображен максимальный поток. Он равен 5.

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. *Басакер, Р.* Конечные графы и сети [Текст] / Р. Басакер, Т. Саати. – М.: Наука, 1974.
2. *Зыков, А.А.* Основы теории графов [Текст] / А.А. Зыков. – М.: Наука, 1987.
3. *Калугин, Н.А.* Основы теории графов[Текст]: учеб. пособие / Н.А. Калугин, А.Н. Калугин. – Самара: Изд-во СГАУ, 2003.
4. *Оре, О.* Теория графов [Текст] / О. Оре. - М.: Мир, 1980.
5. *Уилсон, Р.* Введение в теорию графов [Текст] / Р. Уилсон. – М.: Мир, 1977.
6. *Форд, Л.Р.* Потоки в сетях [Текст] / Л.Р. Форд, Д. Фалкерсон. – М.: Мир, 1966.

Учебное издание

**ОРИЕНТИРОВАННЫЕ
И НЕОРИЕНТИРОВАННЫЕ ГРАФЫ
В ПРИМЕРАХ И ЗАДАЧАХ**

Методические указания

Составители: *Старинова Ольга Леонардовна,
Васильева Ольга Альбертовна,
Баринова Елена Витальевна*

Редактор Т.К. Кретинина
Довёрстка Т.Е. Половнева

Подписано в печать 24.04.2014. Формат 60×84 1/16.
Бумага офсетная. Печать офсетная. Печ. л. 3,5.
Тираж 50 экз. Заказ . Арт. 67/2014.

Самарский государственный аэрокосмический университет.
443086, г. Самара, Московское шоссе, 34.

Изд-во Самарского государственного аэрокосмического университета.
443086 г. Самара, Московское шоссе, 34.