

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению индивидуального задания по дисциплине
«Дискретная математика»

Донецк
2021

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
ДОНЕЦКОЙ НАРОДНОЙ РЕСПУБЛИКИ
ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

КАФЕДРА «КОМПЬЮТЕРНАЯ ИНЖЕНЕРИЯ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению индивидуального задания по дисциплине
«Дискретная математика»

для обучающихся по направлению подготовки
09.03.01 «Информатика и вычислительная техника»
профилей
«Компьютерная инженерия»,
«Системное программирование»,
«Компьютерные системы цифровой экономики»,
«Встроенные компьютерные системы»
всех форм обучения

РАССМОТРЕНО
на заседании кафедры
компьютерной инженерии
Протокол № 3 от 18.11.2011

УТВЕРЖДЕНО
на заседании учебно-издательского
совета ДОННТУ
Протокол № ____ от _____

Донецк
2021

УДК 510.54(076)
ББК 22.176я73
М54

Составитель:

Чередникова Ольга Юрьевна – кандидат технических наук, доцент кафедры компьютерной инженерии ГОУВПО «ДОННТУ»

- Методические указания к выполнению индивидуального задания по дисциплине «Дискретная математика»** [Электронный ресурс]: для обучающихся по направлению подготовки 09.03.01 «Информатика и вычислительная техника» профилей «Компьютерная инженерия», «Системное программирование», «Компьютерные системы цифровой экономики», «Встроенные компьютерные системы» всех форм обучения / ГОУВПО «ДОННТУ», Каф. компьютерной инженерии; сост. О. Ю. Чередникова. — Донецк : ДОННТУ, 2021. – Систем. требования: Acrobat Reader. – Загл. с титул. экрана.

Методические указания разработаны с целью оказания помощи обучающимся в получении практических навыков по дисциплине «Дискретная математика» по теме «Графы». Пособие содержит требования к выполнению индивидуального задания (расчетной работы) и варианты заданий, а также примеры выполнения задания.

УДК 510.54(076)
ББК 22.176я73

Оглавление

ВВЕДЕНИЕ	5
ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ	7
Представление графов.....	7
Определение численных метрических характеристик	8
Матрица достижимости	10
Максимальный поток в сети	11
Эйлеров путь или цикл.....	11
Задача коммивояжера (Гамильтонов путь или цикл)	12
ВАРИАНТЫ ВЫБОРА ГРАФА	14
ПРИМЕР ВЫПОЛНЕНИЯ РАСЧЕТНОЙ РАБОТЫ	25
ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА.....	34

ВВЕДЕНИЕ

Индивидуальное задание по дисциплине «Дискретная математика» представляет собой расчетную работу по теме «Графы». Задания включают в себя ручной расчет основных метрических характеристик графа, а также ручной проход наиболее важных алгоритмов на графах.

Для студентов очной формы обучения выполнение расчетной работы оценивается в 10 баллов.

Для студентов заочной формы обучения расчетная работа является частью контрольной работы и оценивается в 40 баллов.

Отчет оформляется в тетради и является необходимым условием для допуска к экзамену по дисциплине.

ЗАДАНИЯ К РАСЧЕТНОЙ РАБОТЕ

Необходимо выполнить **ручной расчет** для перечисленных в табл. 1 задач применительно к графу в соответствии с вариантом.

Таблица 1 – Задания расчетной работы

№	Задание	Литература
1	Способы хранения информации о графе при программировании (матрицы смежности, инцидентности, список пар, список смежности)	[1:стр. 154-156] [2:стр. 86-89] [3:стр. 107-110]
2	Определение численных метрических характеристик (радиус, диаметр)	[1:стр.163] [3:стр. 148-149] [4:стр. 120-124]
3	Раскраска графа (хроматическое число, хроматический класс)	[1: стр.215] [3: стр.190-193] [4: стр.130-134]
4	Матрица достижимости (для ориентированного графа)	[1: стр.169-170] [2: стр.90] [3: стр.143-144]
5	Нахождение цикломатического числа и минимального остова графа	[1: стр.173-181] [2: стр.135] [3: стр.183-188]
6	Эйлеров цикл (путь)	[1:стр.166] [2: стр.141-142] [3:стр.160-165]
7	Задача коммивояжера (минимальный Гамильтонов путь или цикл)	[1:стр.165] [2: стр.143] [3: стр.165-168]
8	Минимальные расстояния между всеми вершинами графа – ручной проход алгоритма Флойда	[1: стр.202]
9	Максимальный поток в сети	[2: стр.161-167] [3: стр.193-195]

ТЕОРЕТИЧЕСКИЕ СВЕДЕНИЯ

Представление графов

Пусть G -граф с n вершинами X_1, X_2, \dots, X_n и m ребрами U_1, U_2, \dots, U_m . Его матрицей смежности называется квадратная $(n * n)$ матрица $A = (A_{ik})$, где A_{ik} - число ребер, соединяющих вершины X_i и X_k . В частности, $A_{ik} = 0$, если вершины X_i и X_k не смежны. Такая матрица симметрична для неориентированных графов.

Матрицей инциденции (инцидентности) графа G называется $(n * m)$ матрица $B = (B_{ik})$, где $B_{ik} = 1$, если вершина X_i инцидентна ребру U_k и $B_{ik} = 0$ если X_i и U_k не инцидентны. Для графа без петель в каждом столбце матрицы инциденций точно два элемента равны 1. Если граф имеет петли, то в столбцах, соответствующих петлям стоит по одной единице, а в остальных по две.

Более экономным в отношении памяти (особенно в случае неплотных графов $m \ll n^2$) является метод представления графа списком пар, соответствующих его ребрам. Пара $\langle X, Y \rangle$ соответствует дуге $\langle X, Y \rangle$, если граф ориентированный и ребру $\{X, Y\}$ в случае неориентированного графа.

Лучшим решением во многих случаях оказывается структура данных, называемая списком смежности. Он представляет собой массив, i -ый элемент которого содержит список всех вершин, смежных с i -ой.

Каждый элемент такого списка является записью g , содержащей вершину g .строка и указатель g .след на следующую запись в списке (g .след = null(0) для последней записи в списке).

Более простой вариант реализации списка смежности: описать его матрицей $n * n$. Каждая строка матрицы – это список смежных вершин. Признак конца списка – ноль в элементе матрицы.

Например, для графа, изображенного на рис.1, на рис.2 изображены все 4 способа представления.

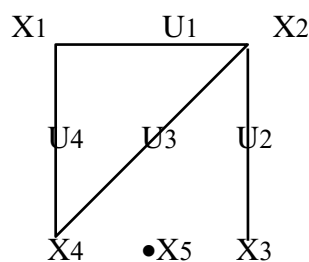


Рисунок 1 – Граф
список пар

матрица смежности	матрица инцидентности	список пар	список смежности																																																																																																					
<table><tr><th></th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th></tr><tr><th>1</th><td>0</td><td>1</td><td>0</td><td>1</td><td>0</td></tr><tr><th>2</th><td>1</td><td>0</td><td>1</td><td>1</td><td>0</td></tr><tr><th>3</th><td>0</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><th>4</th><td>1</td><td>1</td><td>0</td><td>0</td><td>0</td></tr><tr><th>5</th><td>0</td><td>0</td><td>0</td><td>0</td><td>0</td></tr></table> <div>n</div>		1	2	3	4	5	1	0	1	0	1	0	2	1	0	1	1	0	3	0	1	0	0	0	4	1	1	0	0	0	5	0	0	0	0	0	<table><tr><th></th><th>U1</th><th>U2</th><th>U3</th><th>U4</th></tr><tr><th>X1</th><td>1</td><td>0</td><td>0</td><td>1</td></tr><tr><th>X2</th><td>1</td><td>1</td><td>1</td><td>0</td></tr><tr><th>X3</th><td>0</td><td>1</td><td>0</td><td>0</td></tr><tr><th>X4</th><td>0</td><td>0</td><td>1</td><td>1</td></tr><tr><th>X5</th><td>0</td><td>0</td><td>0</td><td>0</td></tr></table> <div>m</div>		U1	U2	U3	U4	X1	1	0	0	1	X2	1	1	1	0	X3	0	1	0	0	X4	0	0	1	1	X5	0	0	0	0	<table><tr><th>ребро</th><th>нач</th><th>конец</th></tr><tr><td>U1</td><td>1</td><td>2</td></tr><tr><td>U2</td><td>2</td><td>3</td></tr><tr><td>U3</td><td>2</td><td>4</td></tr><tr><td>U4</td><td>1</td><td>4</td></tr></table>	ребро	нач	конец	U1	1	2	U2	2	3	U3	2	4	U4	1	4	<table><tr><td>1</td><td>X2</td><td>X4</td><td></td></tr><tr><td>2</td><td>X1</td><td>X3</td><td>X4</td></tr><tr><td>3</td><td>X2</td><td></td><td></td></tr><tr><td>4</td><td>X1</td><td>X2</td><td></td></tr><tr><td>5</td><td></td><td></td><td></td></tr></table>	1	X2	X4		2	X1	X3	X4	3	X2			4	X1	X2		5			
	1	2	3	4	5																																																																																																			
1	0	1	0	1	0																																																																																																			
2	1	0	1	1	0																																																																																																			
3	0	1	0	0	0																																																																																																			
4	1	1	0	0	0																																																																																																			
5	0	0	0	0	0																																																																																																			
	U1	U2	U3	U4																																																																																																				
X1	1	0	0	1																																																																																																				
X2	1	1	1	0																																																																																																				
X3	0	1	0	0																																																																																																				
X4	0	0	1	1																																																																																																				
X5	0	0	0	0																																																																																																				
ребро	нач	конец																																																																																																						
U1	1	2																																																																																																						
U2	2	3																																																																																																						
U3	2	4																																																																																																						
U4	1	4																																																																																																						
1	X2	X4																																																																																																						
2	X1	X3	X4																																																																																																					
3	X2																																																																																																							
4	X1	X2																																																																																																						
5																																																																																																								

Рис.2 – Способы представления графа

Определение численных метрических характеристик

Радиус и диаметр

Путь наименьшей длины, соединяющий вершины x_i и x_j называется расстоянием между вершинами $p(x_i, x_j) = \min_k l(M_{ij}^k)$.

Если x_i и x_j не соединены цепью (т.е. принадлежат разным компонентам), то $P(x_i, x_j) = \infty$.

Наибольшее расстояние между любыми двумя вершинами графа - его **диаметр**:

$$\alpha(G) = \max_{\forall i, j} P(x_i, x_j) = \max_{\forall i, j} \min_k l(M_{ij}^k);$$

Радиус вершины x_i : $r(x_i) = \max_{\forall j} P(x_i, x_j)$;

Радиус графа - минимальный из радиусов вершин:

$$r(G) = \min_{\forall i} r(x_i) = \min_{\forall i} \max_{\forall j} P(x_i, x_j);$$

Для более наглядного поиска радиуса и диаметра графа сначала составляют матрицу расстояний. Максимальное значение в каждой строке является радиусом соответствующей вершины. Максимальное значение в матрице – это диаметр графа, а минимальное значение среди радиусов вершин является радиусом графа.

Например, определим $D(G)$ и $R(G)$ графа (рис.3):

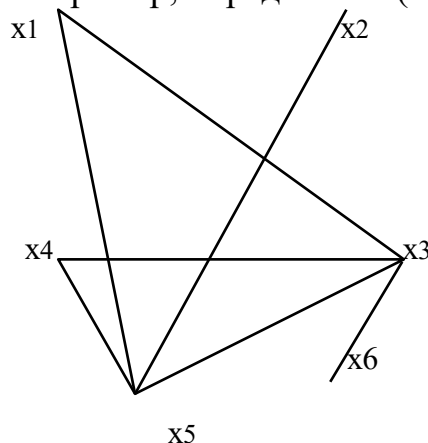


Рисунок 3 – Пример графа для нахождения радиуса и диаметра

1) Сначала построим матрицу расстояний:

	1	2	3	4	5	6
1	0	2	1	2	1	2
2	2	0	2	2	1	3
3	1	2	0	1	1	1
4	2	2	1	0	1	2
5	1	1	1	1	0	2
6	2	3	1	2	2	0

Матрица показывает кратчайшие пути (расстояния) из i -ой вершины в j -ую.

2) Максимальный элемент в каждой строке (столбце) – это радиус i -ой вершины:

вершина	1	2	3	4	5	6
радиус	2	3	2	2	2	3

3) Максимальный из радиусов вершин – это диаметр графа:

$$\alpha(G) = 3 = \max_{\forall i, j} P(x_i, x_j).$$

Минимальный из радиусов вершин – это радиус графа:

$$r(G) = \min_i \max_j P(x_i, x_j) = 2.$$

Хроматическое число и класс (раскраска) графа

Каждую вершину графа (соответственно каждое ребро) можно раскрасить в какой-либо цвет. Раскраска вершин (ребер) графа называется правильной, если никакие 2 смежные вершины (2 смежных ребра) не окрашены в один и тот же цвет. Наименьшее число цветов p которыми можно раскрасить вершины графа, называется **хроматическим числом** графа.

Например (рис.4):

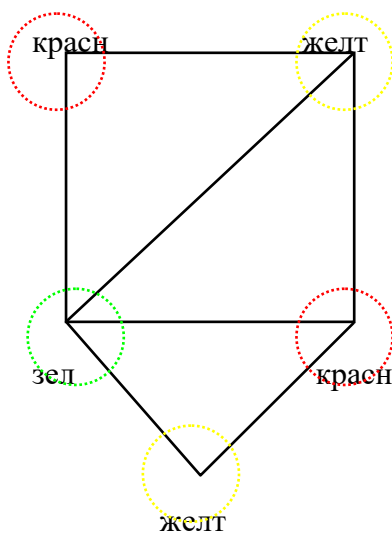


Рисунок 4 – Раскраска графа

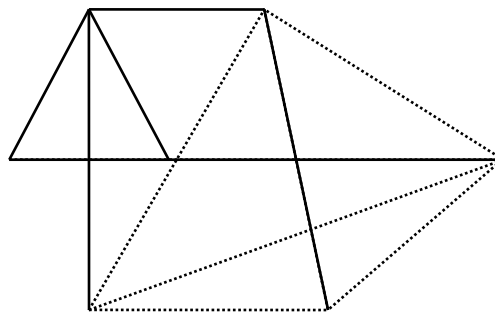


Рисунок 5 – Остов графа

Хроматическое число = 3

Наименьшее число $q = w(G)$ цветов достаточных для правильной раскраски ребер графа G называется **хроматическим классом**.

Цикломатическое число и остов графа

Пусть G - произвольный граф с n вершинами, m ребрами и p компонентами связности. Величина $V = V(G) = m - n + p$, $V(G) \geq 0$ называется **цикломатическим числом** графа G .

Цикломатическое число графа G равно количеству ребер, которое нужно удалить из G , не изменяя p , чтобы превратить G в граф без циклов (построить **остов** графа).

На рис. 5 показано построение остова (сплошными линиями и ребра, которые были удалены – пунктирными линиями). Цикломатическое число этого графа $11 - 7 + 1 = 5$, что совпадает с количеством удаленных ребер.

Матрица достижимости

Вершина x_j называется достижимой на x_i , если существует путь $x_i \Rightarrow x_j$. Аналогично матрице, задающей граф, можно построить матрицы достижимости

$$P: P_{ij} = \begin{cases} 1, & \text{if } \exists x_i \rightarrow x_j \\ 0, & \text{else} \end{cases}$$

Минимальные расстояния между всеми вершинами графа

В работе предполагается использование алгоритма Флойда, который находит кратчайшие расстояния от каждой вершины до каждой во взвешенном графе. Ответом является матрица кратчайших расстояний. Исходными данными является начальная матрица расстояний MR, которая заполняется следующим образом: если вершины смежны, в ячейке ставится вес ребра, иначе – бесконечность. Блок-схема метода приведена на рис. 6.

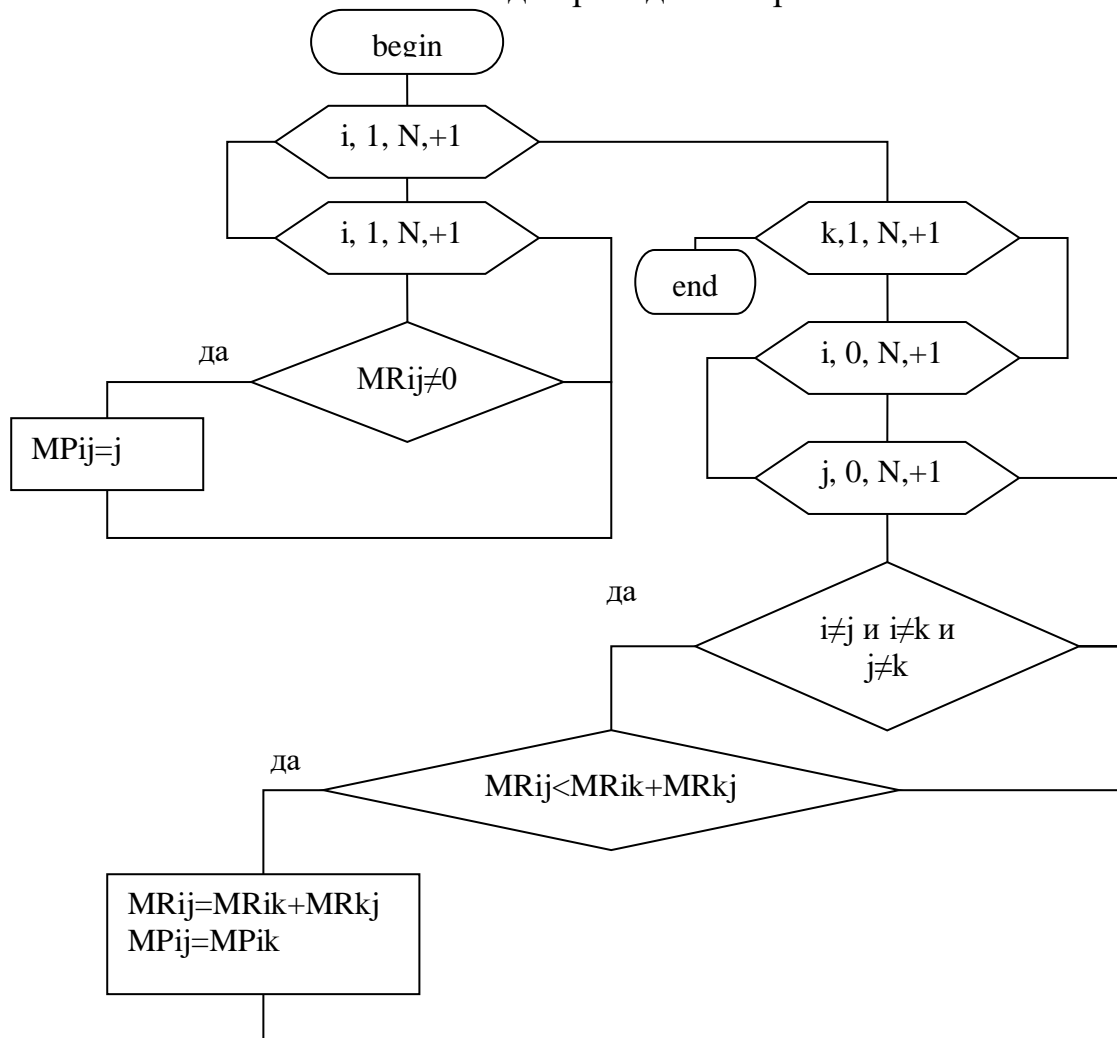


Рисунок 6 – Блок-схема алгоритма Флойда

Ответом является матрица расстояний и матрица пути MP. По матрице пути можно восстановить маршрут. Каждый элемент содержит первый узел в маршруте.

Максимальный поток в сети

Для выполнения задания граф нужно ориентировать, чтобы он удовлетворял условиям сети:

- 1) существует одна и только одна вершина x_0 (вход) в которую не заходит ни одна дуга (исток);
- 2) существует одна и только одна вершина x_N (выход) = Z из которой не выходит ни одна дуга (сток).
- 3) каждая дуга имеет вес, который называется пропускной способностью дуги $C(U)$.

Для сети необходимо найти максимальный поток.

Поток дуги – это величина $F(U)$, удовлетворяющая условиям:

- 1) $0 \leq F(U) \leq C(U)$
- 2) Для каждой вершины сумма потоков дуг, входящих в вершину, всегда равна сумме потоков дуг, выходящих из вершины.

Дуга называется насыщенной, если для нее $F(U) = C(U)$.

На первом этапе решения задачи необходимо выполнить начальную расстановку потоков дуг согласно перечисленным выше правилам. Самый простой вариант – принять начальный поток равным нулю для каждой дуги.

Второй этап – этап насыщения. Предполагает следующие действия:

1. Находим любой маршрут, начинающийся в истоке и заканчивающийся в стоке и не содержащий насыщенных дуг, совпадающих с направлением движения. Если такого маршрута нет, то этап насыщения закончен, иначе – на шаг 2.
2. Для каждой дуги этого маршрута находим величину $q(U) = C(U) - F(U)$, если направление дуги совпадает с направлением движения и $q(U) = F(U)$ иначе.
3. Выбираем минимальное значение $q(U)$.
4. Увеличиваем значение потока каждой дуги выбранного маршрута, совпадающей с его направлением, на величину, найденную на шаге 3. и уменьшаем значение потока на эту величину на дугах, направление которых противоположно направлению маршрута.
5. Переходим на шаг 1.

После этапа насыщения получаем ответ задачи - максимальный поток равен сумме потоков истока, соответственно он должен совпадать с суммой потоков стока.

Эйлеров путь или цикл

Эйлеровым циклом на графе называется цикл, содержащий все ребра и все вершины графа. Такие циклы существуют не на любом графе. Граф содержащий эйлеров цикл называется эйлеровым.

Теорема: Для того, чтобы граф был эйлеровым, необходимо и достаточно, чтобы он был связным (соединены все вершины графа) и все его вершины имели четную степень (эйлерова линия при ее проведении входит в каждую вершину и выходит из нее одно и то же число раз).

Алгоритм нахождения Эйлера цикла:

1. Проверяем является ли граф связным и степени всех его вершин четны;

2. Пусть прокладка начинается в вершине x_1 . От начальной вершины переходим к любой следующей, всякий раз выбирая такое ребро, по которому еще не двигались.

3. Т.к. число ребер в графе конечно, то через какое-то число шагов прокладка будет завершена, а поскольку к каждой вершине подходит столько же ребер, сколько и выходит, то прокладка завершится в начальной вершине.

Может оказаться, что построенный по алгоритму цикл не является эйлеровым, т.е. некоторые вершины не были пройдены (а это значит, что не были пройдены некоторые ребра). Если это так, то на графе необходимо найти вершину x_k из которой выходит ребро, не использованное при прокладке. В этом случае надо совершить дополнительную прокладку, начав ее из вершины x_k и так как $p(x_k)$ -четное, завершив в x_k . После дополнительной прокладки получаем единый цикл, включающий в себя ребра основного и дополнительного циклов. Достигнув вершины x_k нужно перейти на прокладку дополнительного цикла, который завершится в вершине x_k . После этого продолжается прокладка основного цикла. Т.о. замкнутую фигуру, в которой все вершины четные можно начертить без повторений одним росчерком пера.

В расчетной работе Эйлеров цикл показывается перечислением вершин в порядке обхода графа или нумерацией ребер на графе в порядке их обхода. Если условие существования Эйлера цикла не соблюдены, то в РГР необходимо добавить ребра, чтобы условие выполнялось.

Задача коммивояжера (Гамильтонов путь или цикл)

Гамильтонова цепь - простая цепь, проходящая через все его вершины.

Гамильтонов цикл - простой цикл, проходящий через все его вершины.

Задача коммивояжера решается на взвешенных графах и заключается в нахождении минимального гамильтонового цикла (сумма весов минимальна). Удалось создать метод решения, в соответствие с которым шаг за шагом строится не полное, а усеченное граф-дерево. Этот метод называется методом ветвей и границ. Его алгоритм:

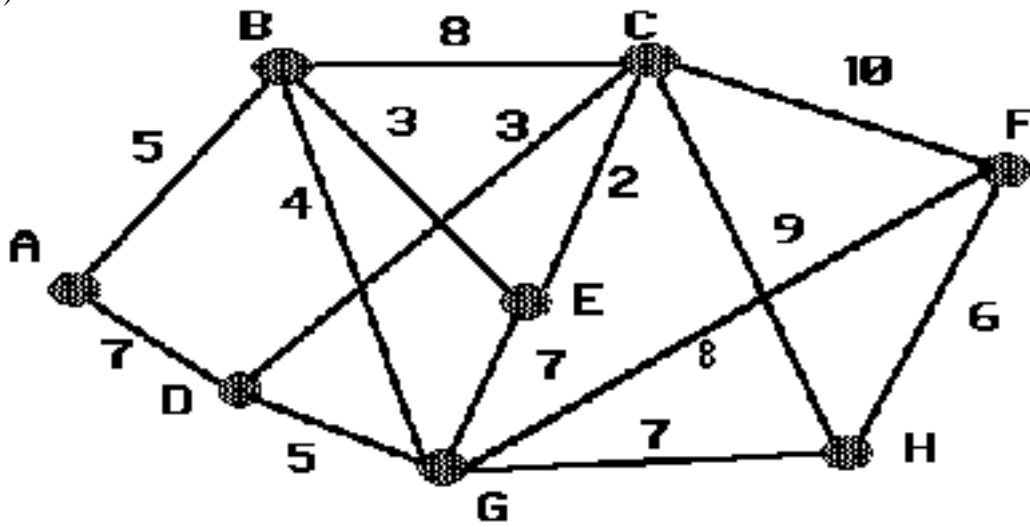
1. Принимает какую-либо вершину за начальную (это корень дерева).
2. Добавляем в дерево вершины, смежные с выбранной и соединяем их ребром с выбранной вершиной. При этом учитываем, что выбирать вершины, через которые мы уже проходили в этой ветви мы не можем. Если все вершины графа были пройдены в ветви, то возвращаемся в начальную вершину – это гамильтонов цикл – и отсекаем вершины графа, оценка которых больше найденной. Если вершин, удовлетворяющих требованиям нет, то это тупик (далее этот путь не рассматриваем).

3. Делаем оценку этих вершин (добавляем вес ребра к оценке выбранной вершины).
4. Выбираем вершину с минимальной оценкой из всех концевых вершин графа.
5. Если у нас есть вершины, которые являются не тупиками, не отсеченными и не конечными вершинами цикла, то переходим на пункт 2. Иначе дерево построено, цикл найден или его нет в графе.

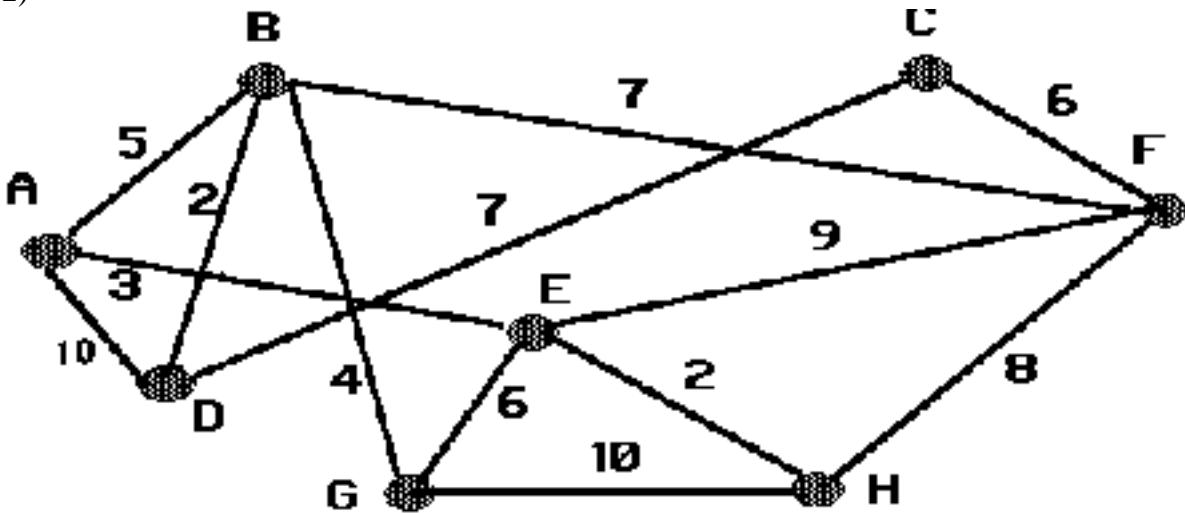
В расчетной работе для выполнения этого задания необходимо оставить в графе 5 вершин (например, находящихся слева).

ВАРИАНТЫ ВЫБОРА ГРАФА

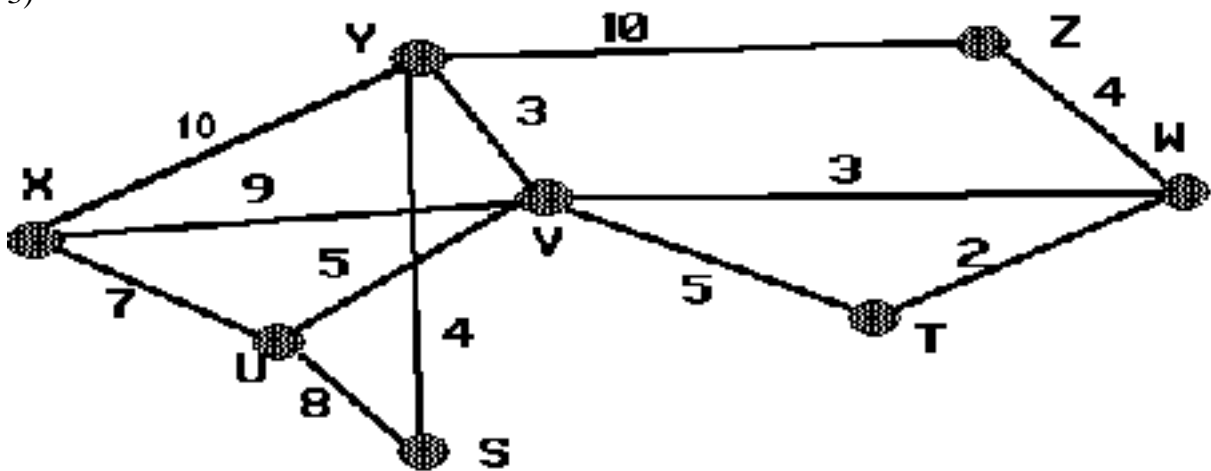
1)



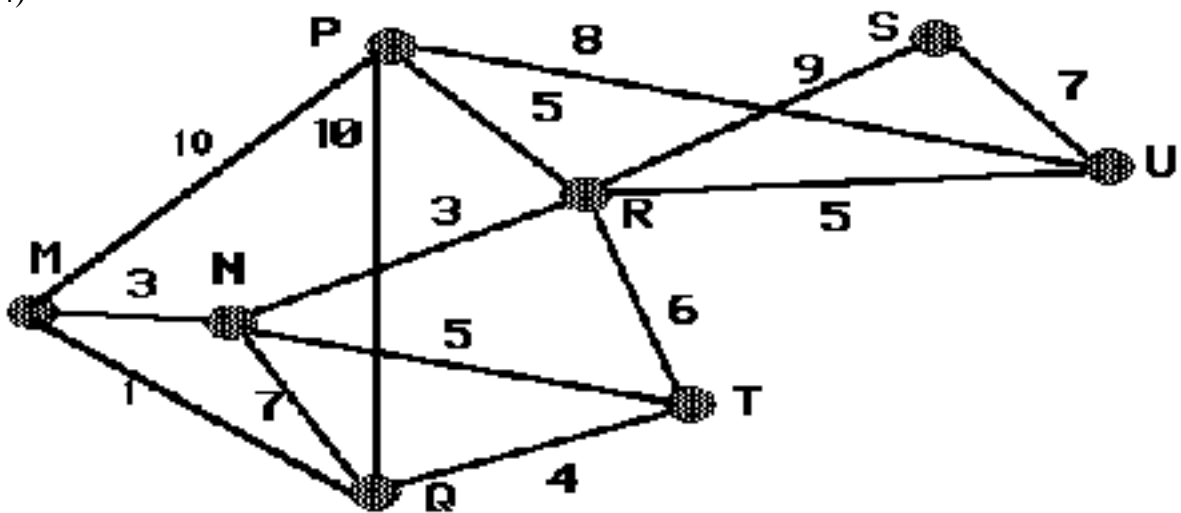
2)



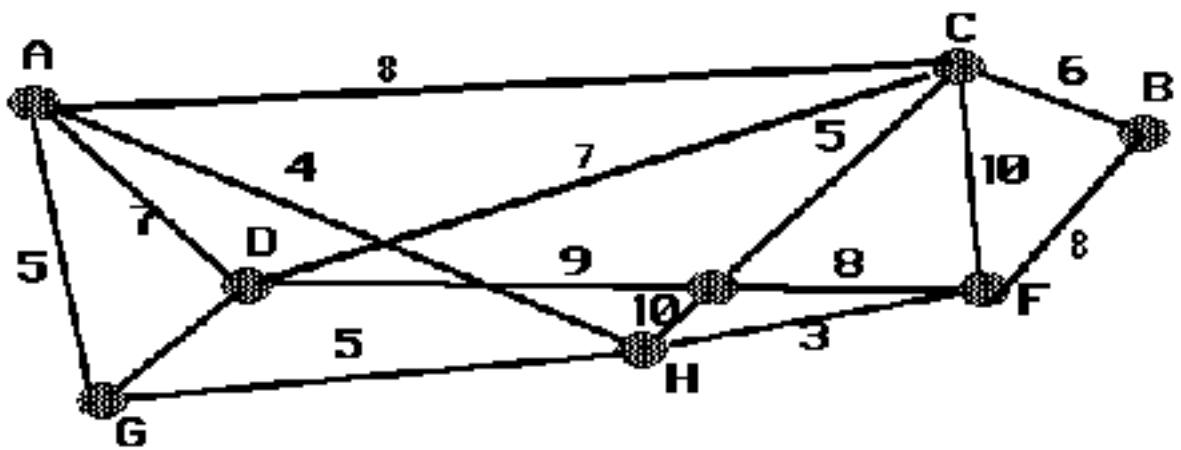
3)



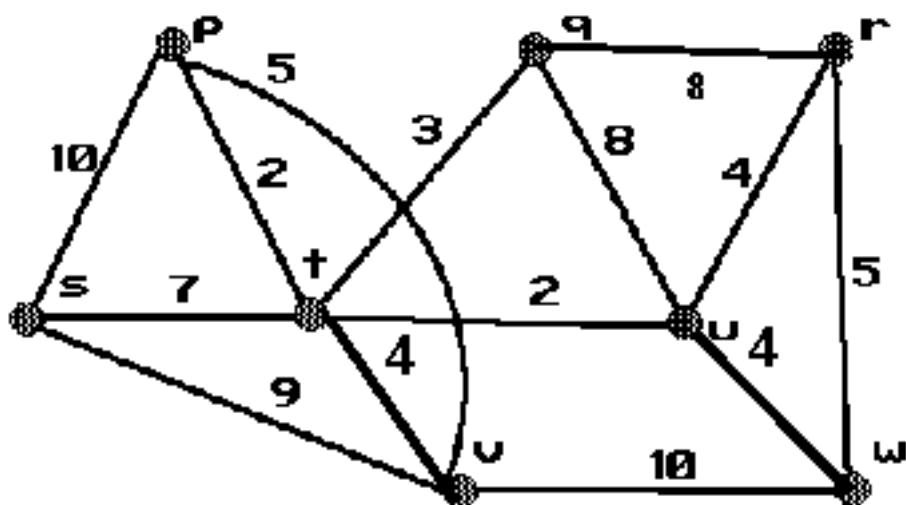
4)



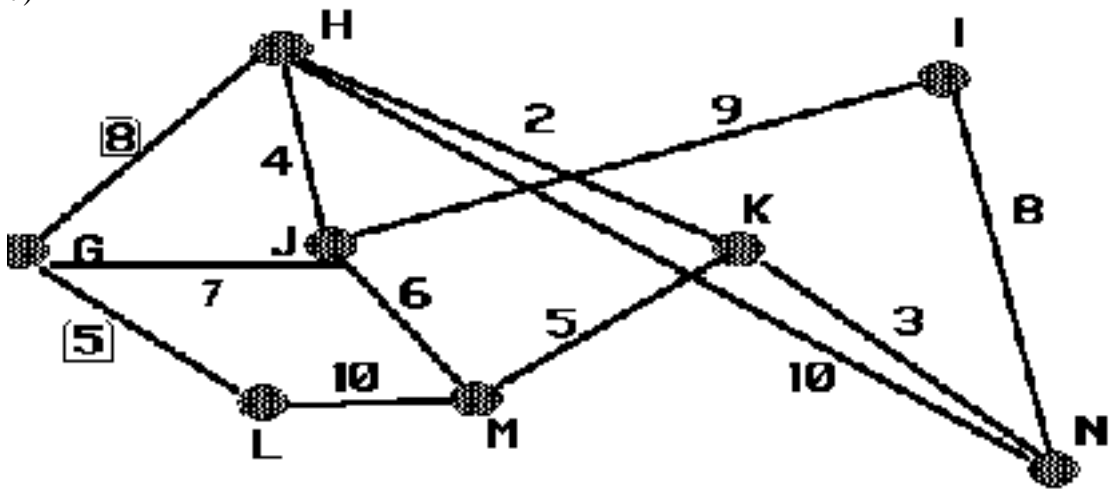
5)



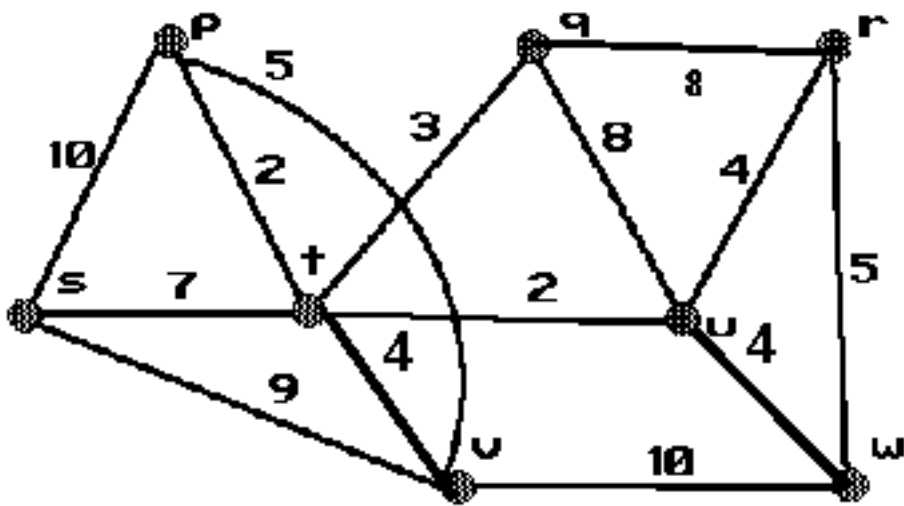
6)



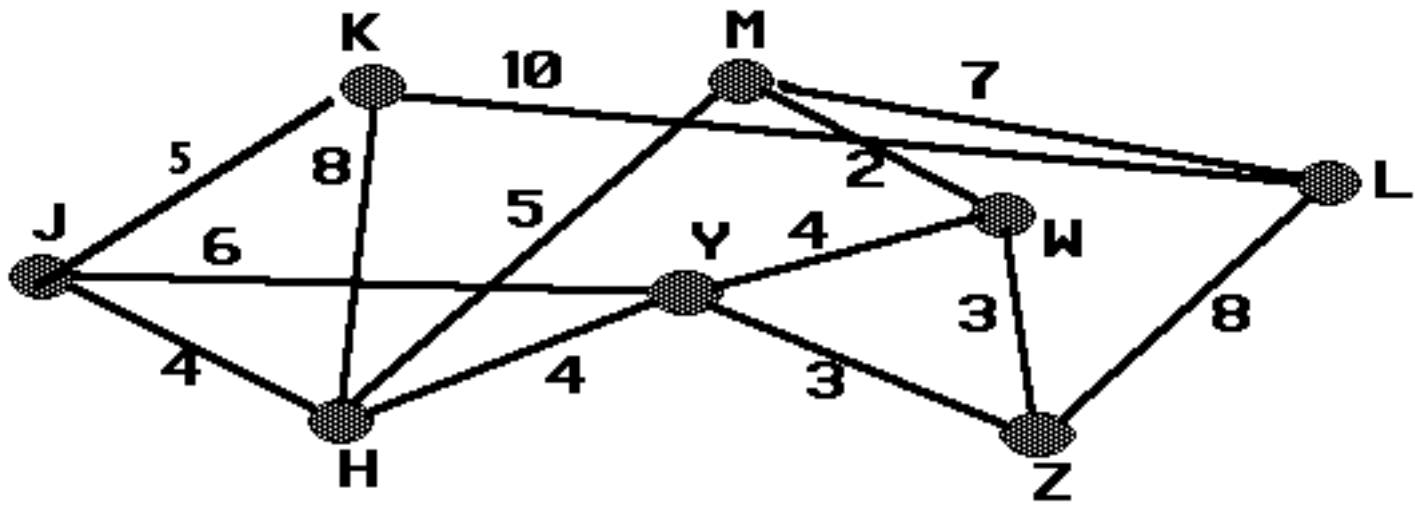
7)



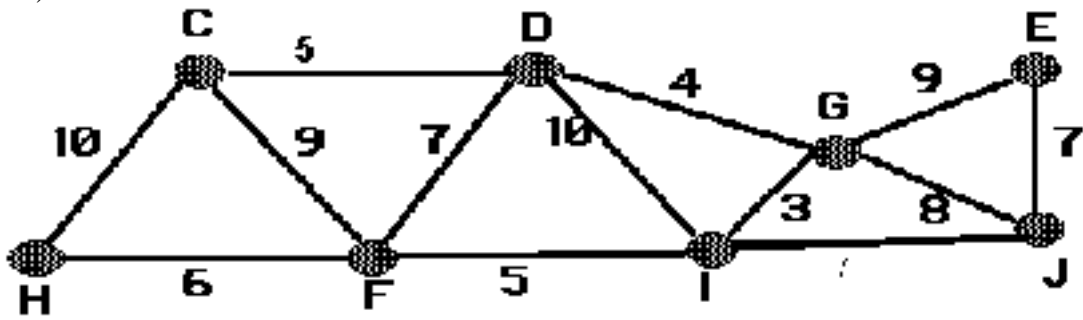
8)



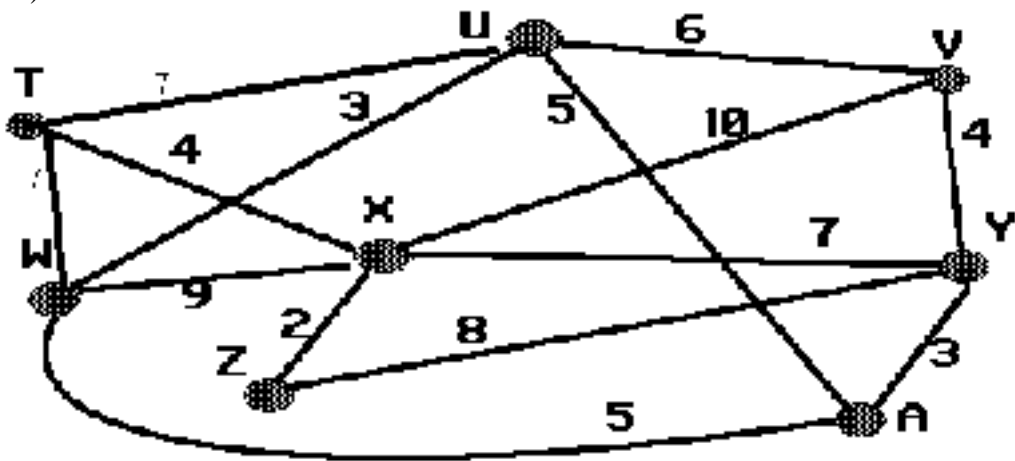
9)



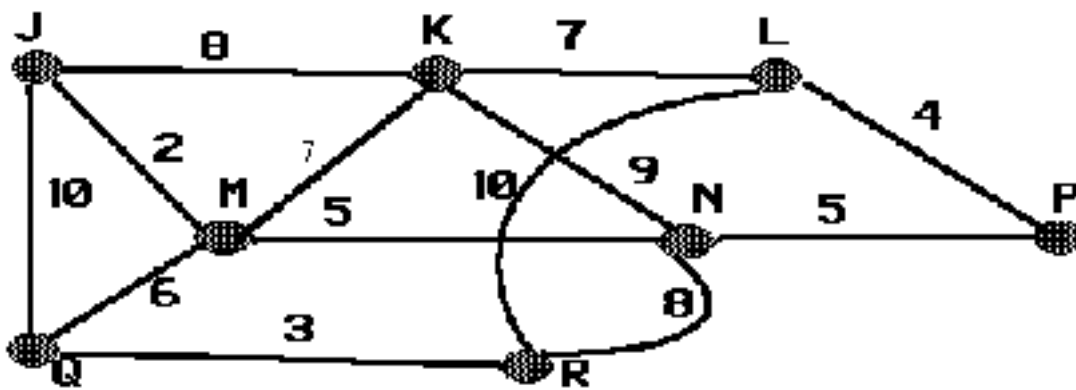
10)



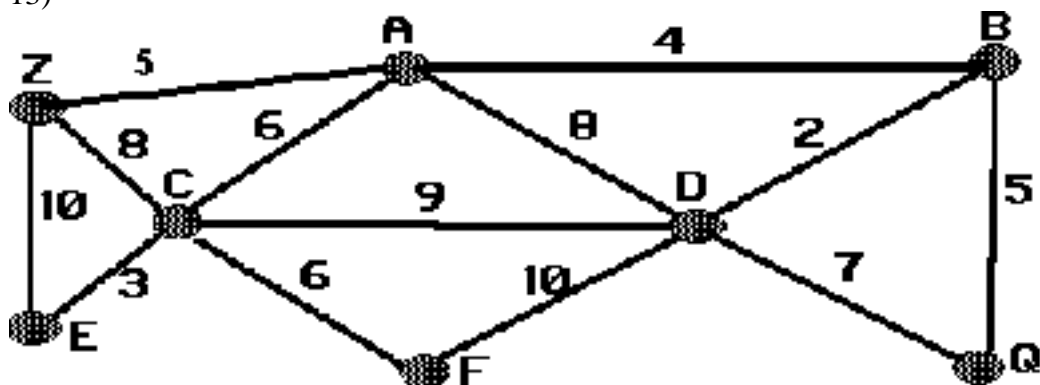
11)



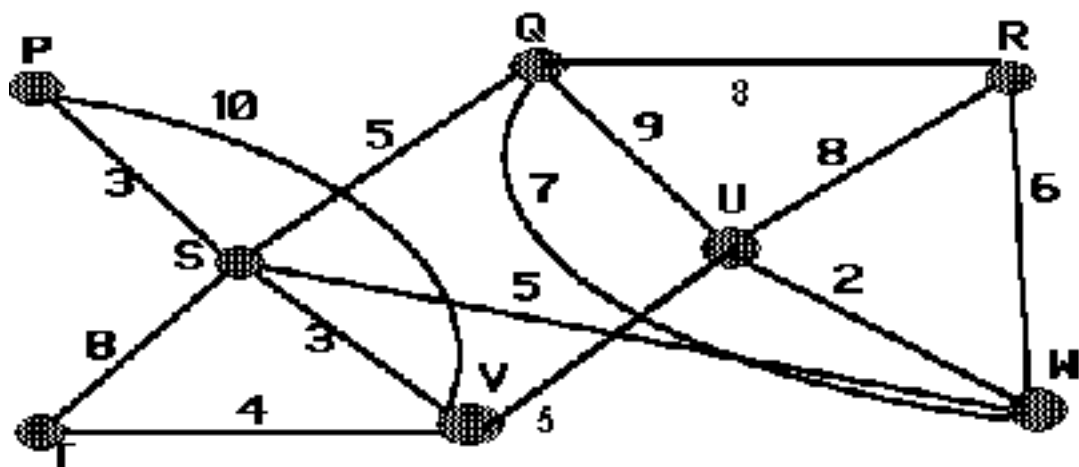
12)



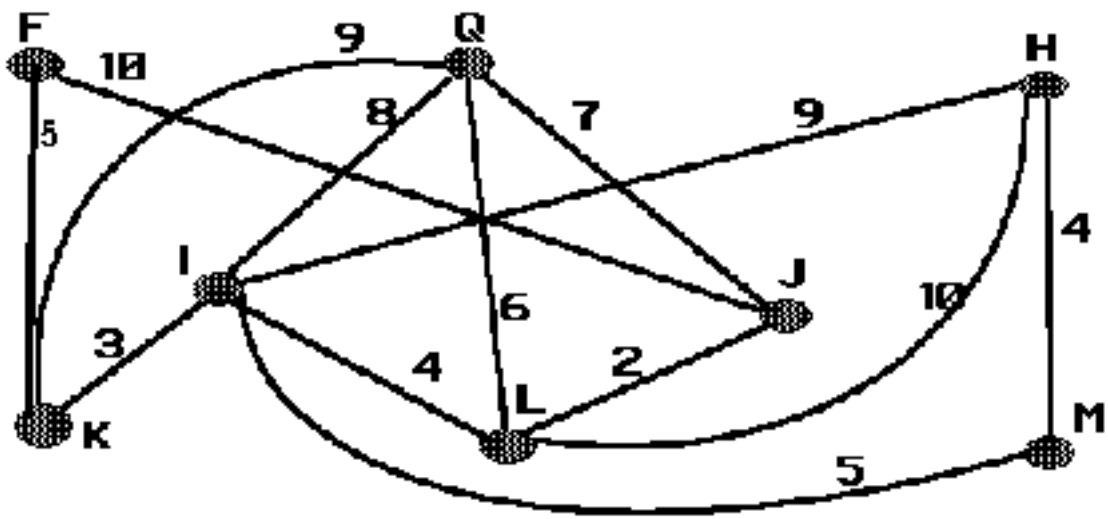
13)



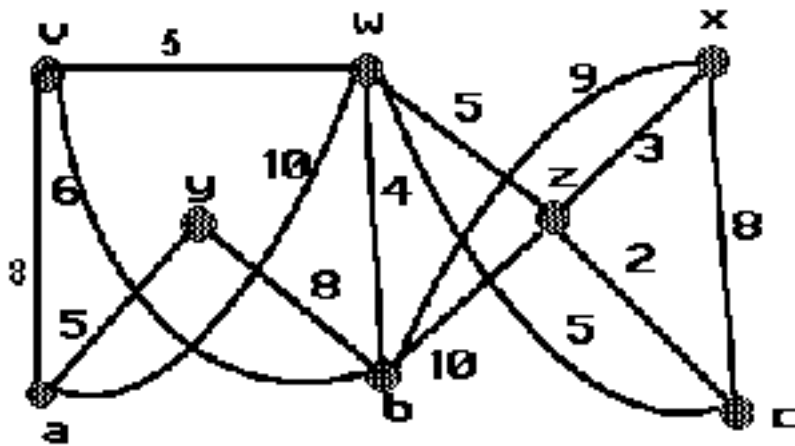
14)



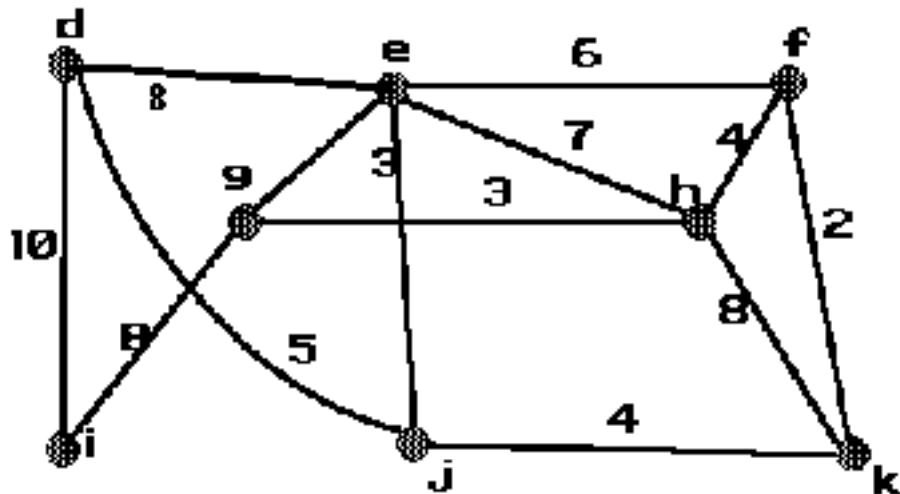
15)



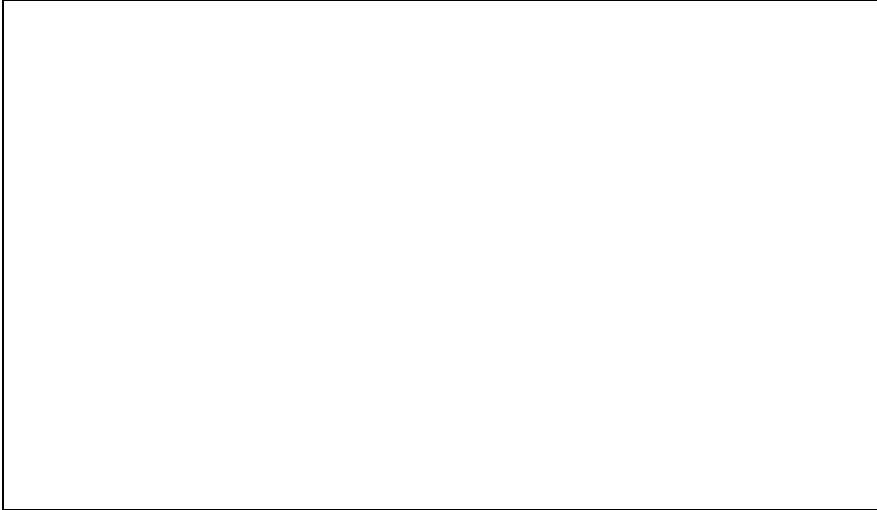
16)



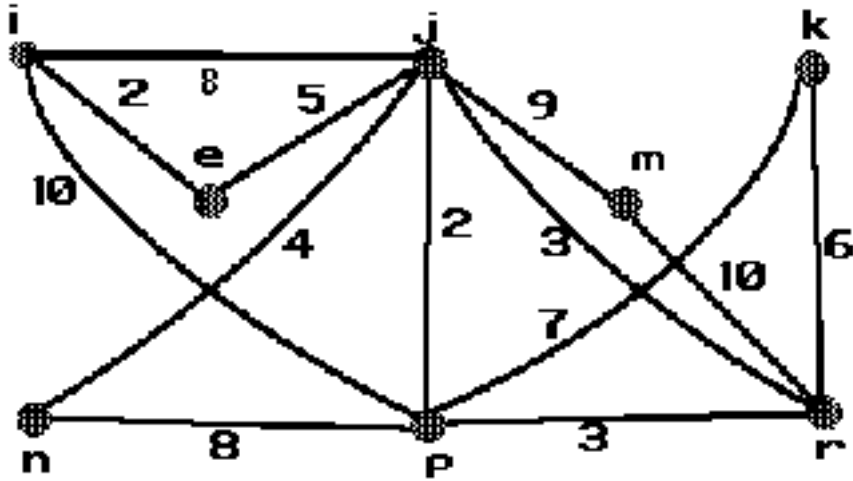
17)



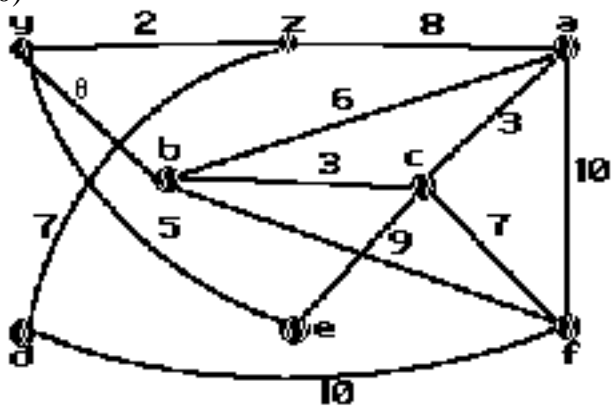
18)



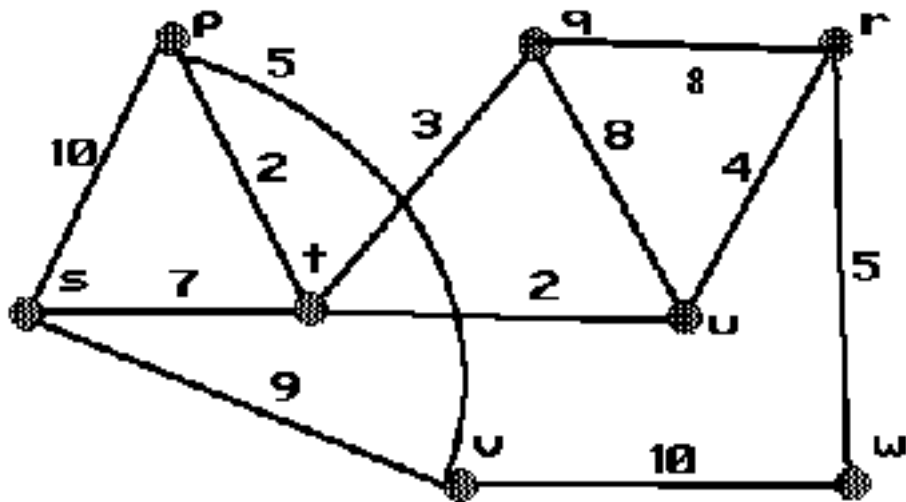
19)



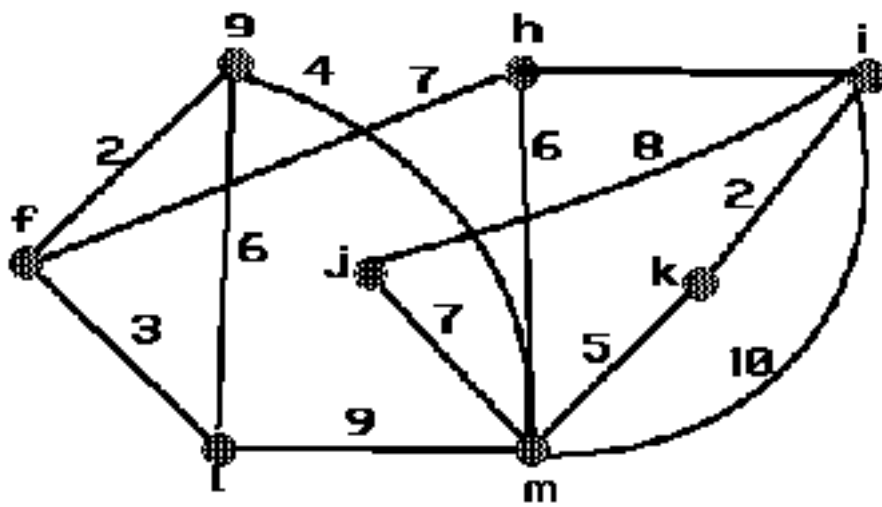
20)



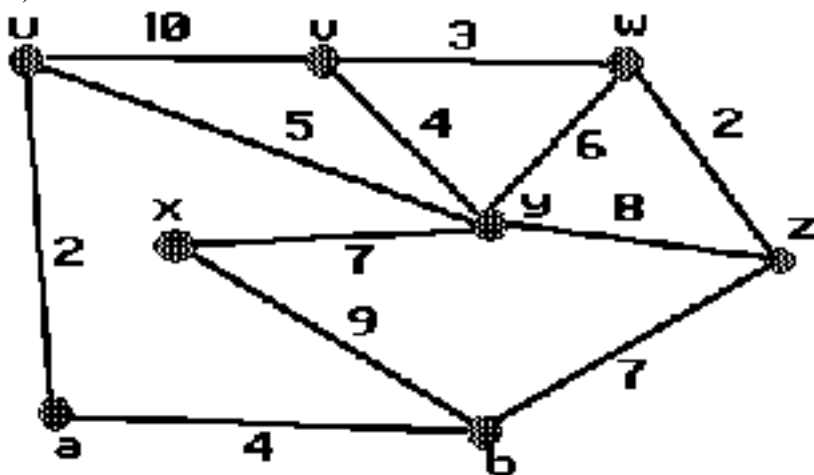
21)



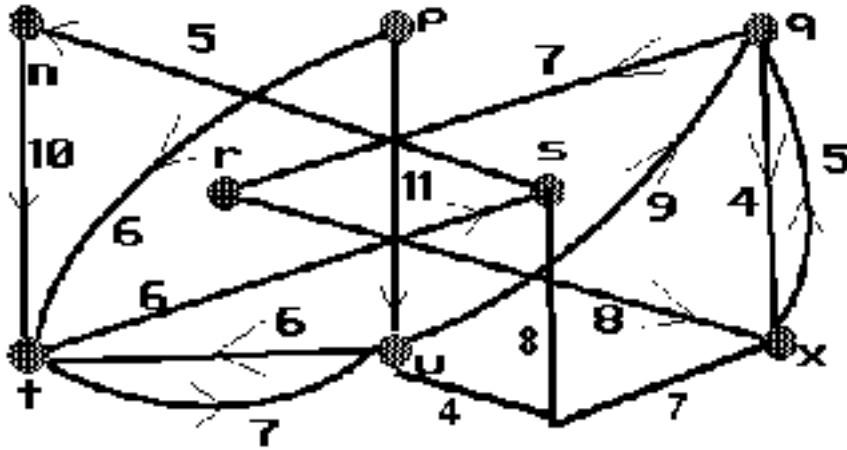
22)



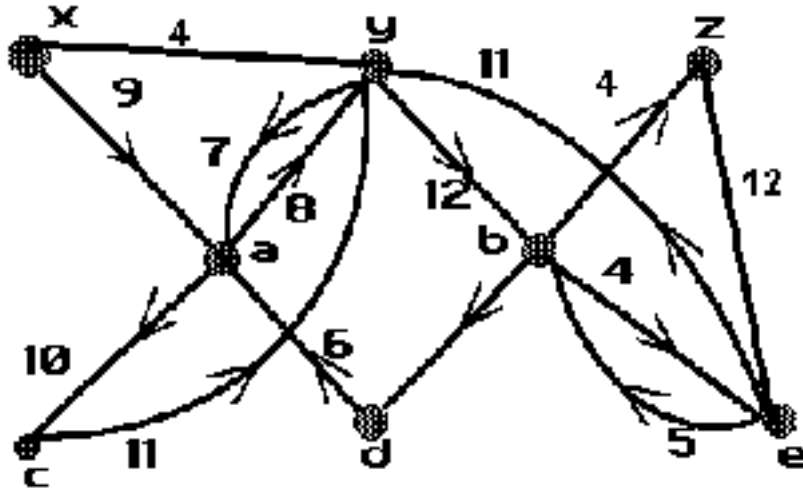
23)



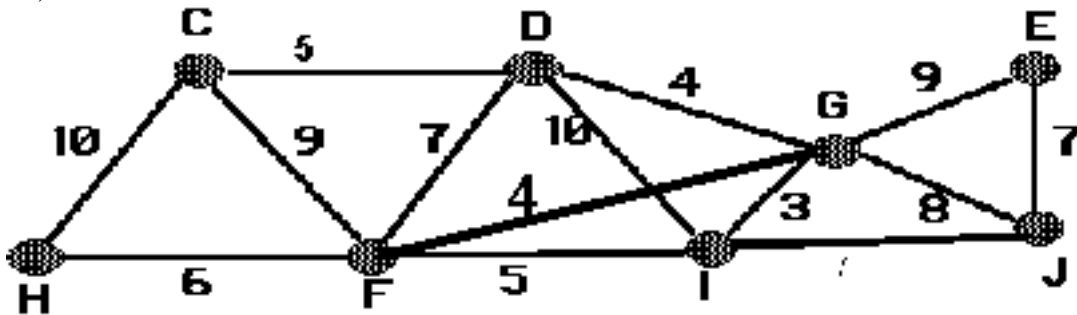
24)



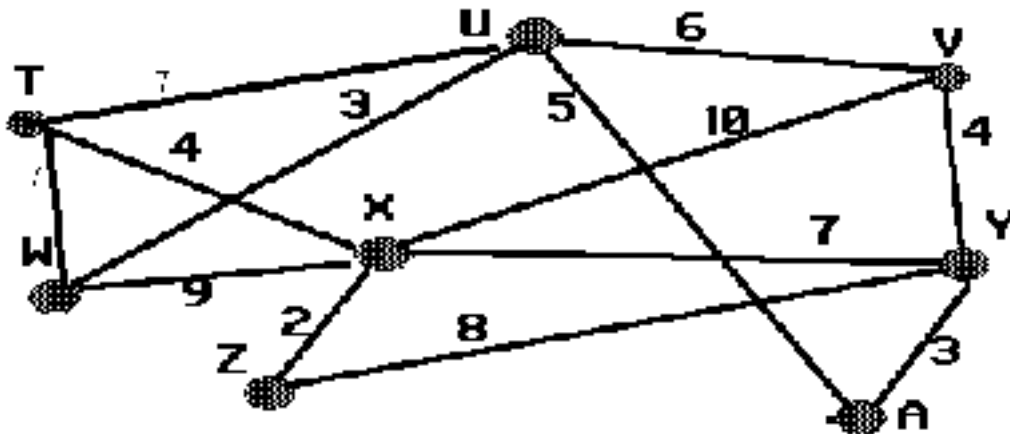
25)



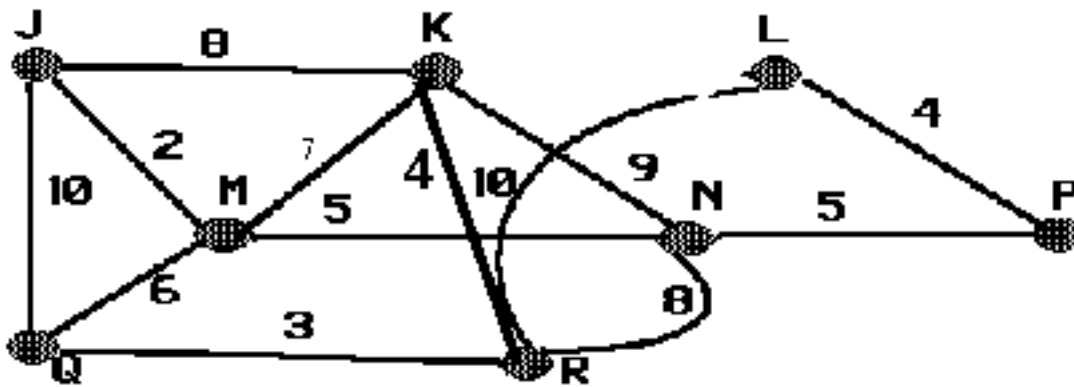
26)



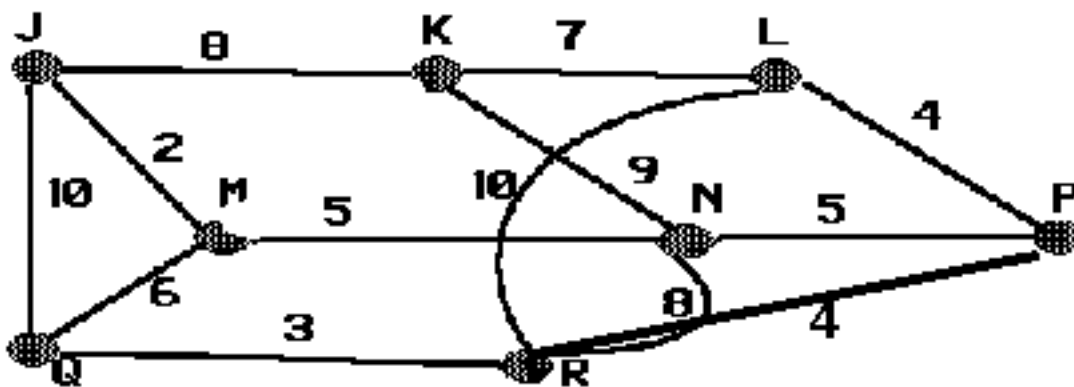
27)



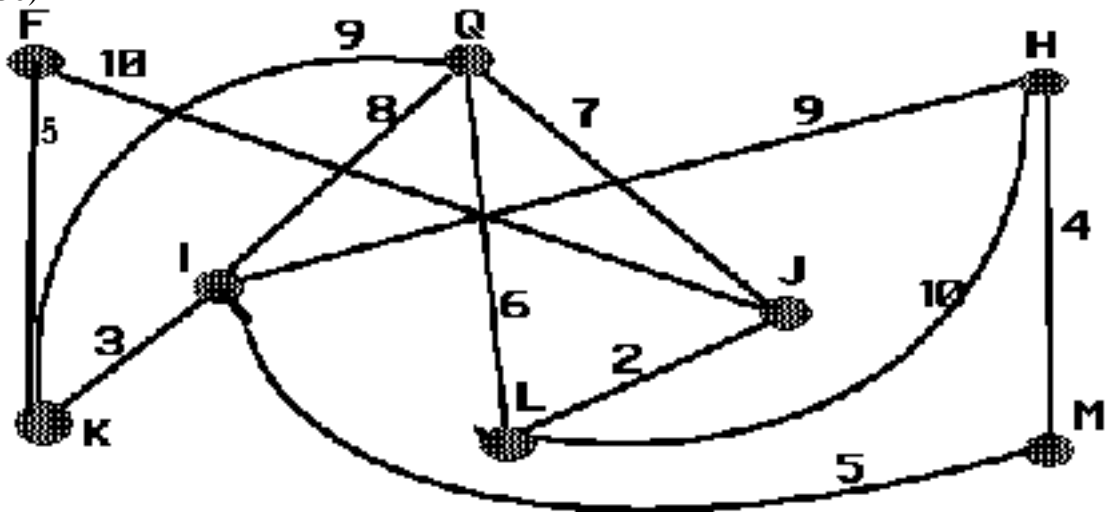
28)



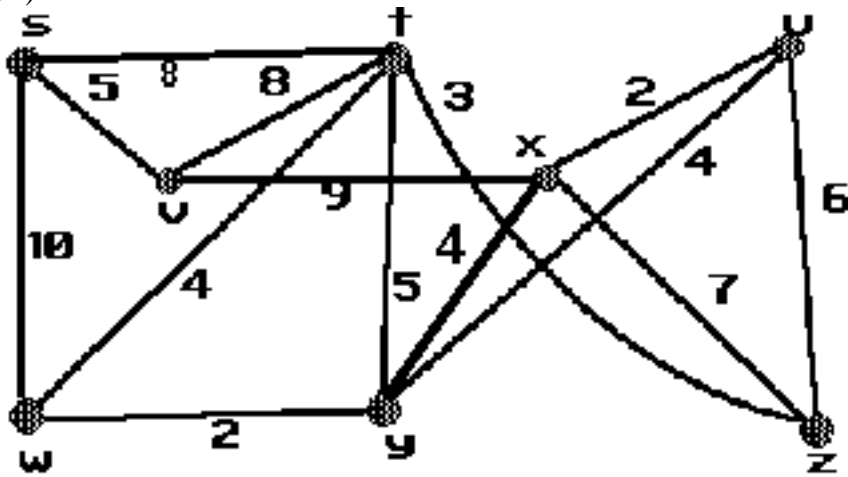
29)



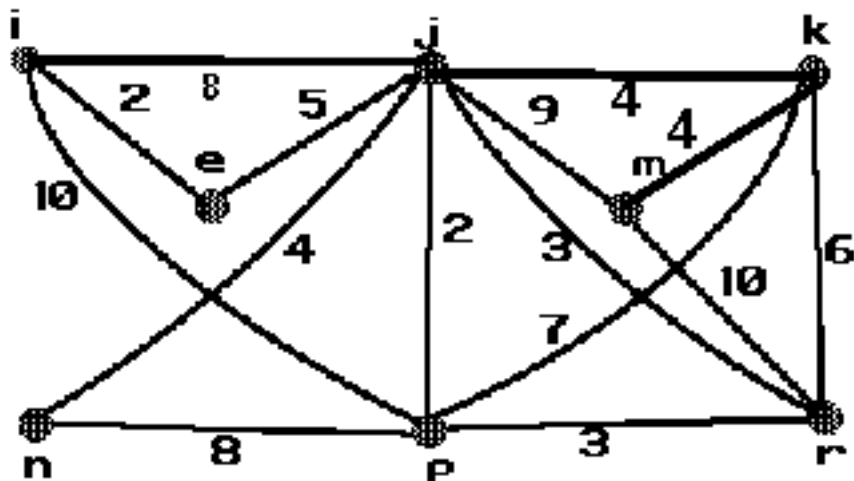
30)



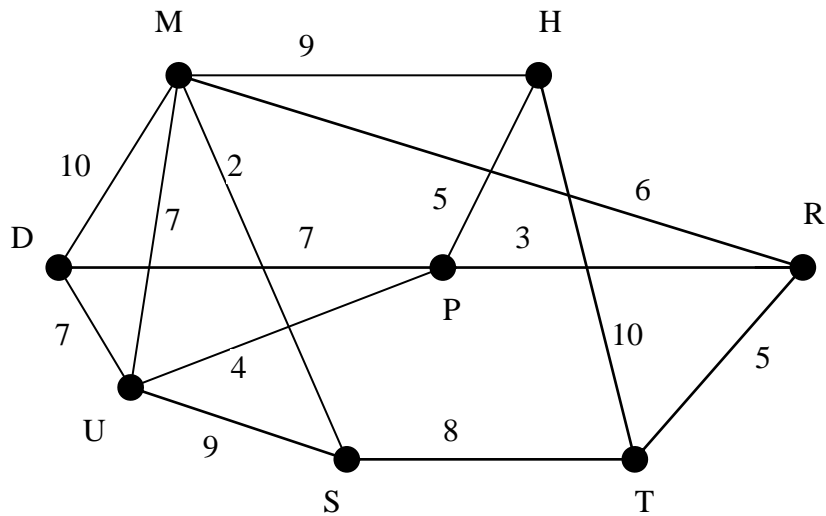
31)



32)



ПРИМЕР ВЫПОЛНЕНИЯ РАСЧЕТНОЙ РАБОТЫ



1. Способы хранения информации о графе

Матрица смежности:

	D	M	U	H	P	S	T	R
D	0	1	1	0	1	0	0	0
M	1	0	1	1	0	1	0	1
U	1	1	0	0	1	1	0	0
H	0	1	0	0	1	0	1	0
P	1	0	1	1	0	0	0	1
S	0	1	1	0	0	0	1	0
T	0	0	0	1	0	1	0	1
R	0	1	0	0	1	0	1	0

Матрица инцидентностей:

	DM	DU	DP	MU	MS	MR	MH	UP	US	ST	PH	PR	TR	TH
D	1	1	1	0	0	0	0	0	0	0	0	0	0	0
M	1	0	0	1	1	1	1	0	0	0	0	0	0	0
U	0	1	0	1	0	0	0	1	1	0	0	0	0	0
H	0	0	0	0	0	0	1	0	0	0	1	0	0	1
P	0	0	1	0	0	0	0	1	0	0	1	1	0	0
S	0	0	0	0	1	0	0	0	1	1	0	0	0	0
T	0	0	0	0	0	0	0	0	0	1	0	0	1	1
R	0	0	0	0	0	1	0	0	0	0	0	1	1	0

Список пар

начало	D	D	D	M	M	M	M	U	U	S	P	P	T	T
--------	---	---	---	---	---	---	---	---	---	---	---	---	---	---

конец	М	U	Р	U	S	R	Н	Р	S	T	Н	R	R	Н
-------	---	---	---	---	---	---	---	---	---	---	---	---	---	---

Список смежности:

D	M	P	U		
M	D	U	S	R	Н
U	D	M	P	S	
Н	M	P	T		
P	D	U	Н	R	
S	U	M	T		
T	S	Н	R		
R	M	P	T		

2. Определение метрических характеристик графа:

Количество вершин: 8.

Количество ребер: 14.

Степени вершин:

D	M	U	Н	P	S	T	R
3	5	4	3	4	3	3	3

Расстояния между вершинами:

	D	M	U	Н	P	S	T	R
D	0	1	1	2	1	2	3	2
M	1	0	1	1	2	1	2	1
U	1	1	0	2	1	1	2	2
Н	2	1	2	0	1	2	1	2
P	1	2	1	1	0	2	2	1
S	2	1	1	2	2	0	1	2
T	3	2	2	1	2	1	0	1
R	2	1	2	2	1	2	1	0

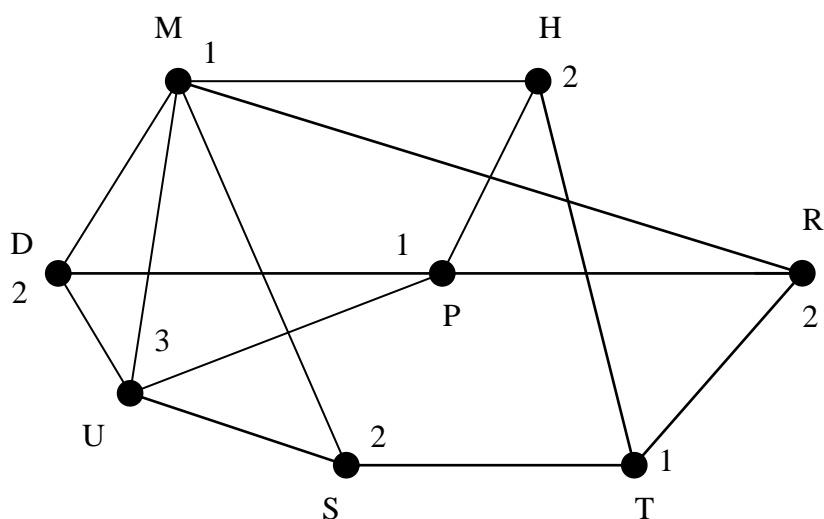
Радиусы вершин:

D	M	U	Н	P	S	T	R
3	2	2	2	2	2	3	2

Диаметр графа: 3.

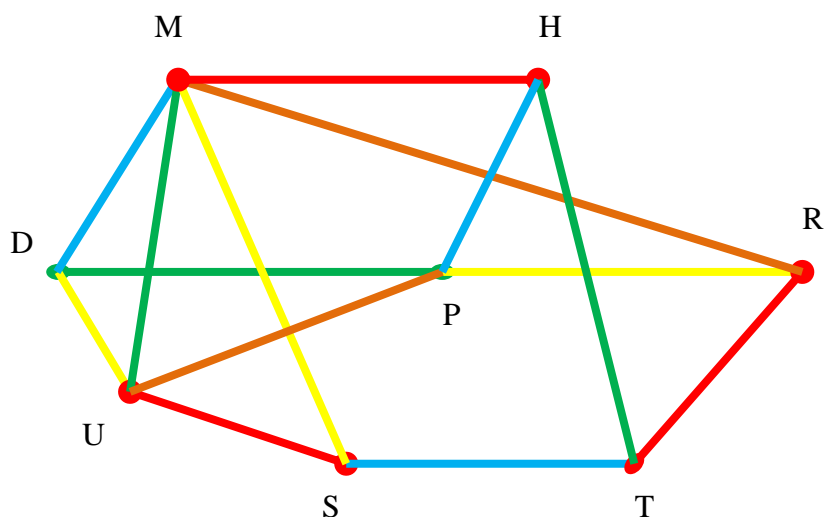
Радиус графа: 2.

Определение хроматического числа графа. Раскраска вершин:



Хроматическое число графа 3.

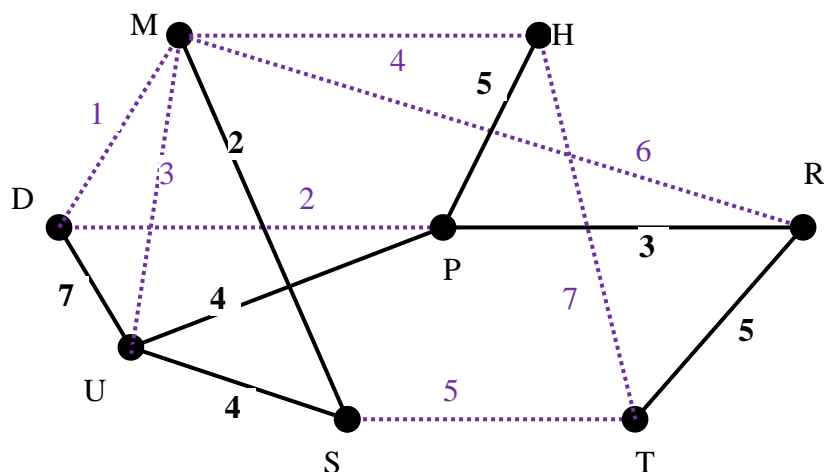
Определение хроматического класса графа. Раскраска ребер:



Хроматический класс графа 5.

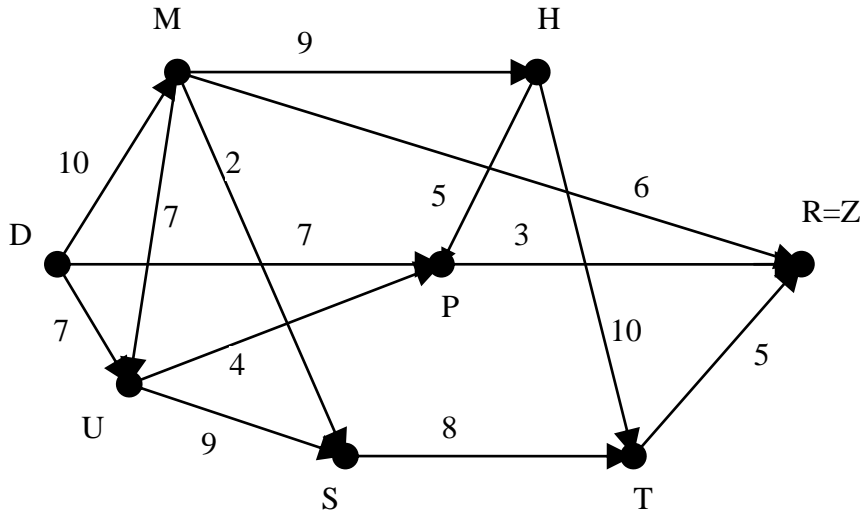
Определение **цикломатического числа**: 1) $14 - 8 + 1 = 7$;

2) **минимальный остов** графа: сумма весов = $2 + 3 + 4 + 4 + 5 + 5 + 7 = 30$



Матрица достижимости:

Ориентированный граф (ориентировали произвольным образом):



	D	M	U	H	P	S	T	R
D	0	1	1	1	1	1	1	1
M	0	0	1	1	1	1	1	1
U	0	0	0	0	1	1	1	1
H	0	1	0	0	1	0	1	1
P	0	0	0	0	0	0	0	1
S	0	0	0	0	0	0	1	1
T	0	0	0	0	0	0	0	1
R	0	0	0	0	0	0	0	0

Алгоритм Флойда поиска минимальных расстояний в графе.

Первичная матрица расстояний

матрица пути

	D	M	U	H	P	S	T	R
D	0	10	7	∞	7	∞	∞	∞
M	10	0	7	9	∞	2	∞	6
U	7	7	0	∞	4	9	∞	∞
H	∞	9	∞	0	5	∞	10	∞
P	7	∞	4	5	0	∞	∞	3
S	∞	2	9	∞	∞	0	8	∞
T	∞	∞	∞	10	∞	8	0	5
R	∞	6	∞	∞	3	∞	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	0	P	0	0	0
M	D	0	U	H	0	S	0	R
U	D	M	0	0	P	S	0	0
H	0	M	0	0	P	0	T	0
P	D	0	U	H	0	0	0	R
S	0	M	U	0	0	0	T	0
T	0	0	0	H	0	S	0	R
R	0	M	0	0	P	0	T	0

Работа алгоритма.

Через вершину D:

	D	M	U	H	P	S	T	R		D	M	U	H	P	S	T	R
D	0	10	7	∞	7	∞	∞	∞	D	0	M	U	0	P	0	0	0
M	10	0	7	9	17	2	∞	6	M	D	0	U	H	D	S	0	R
U	7	7	0	∞	4	9	∞	∞	U	D	M	0	0	P	S	0	0
H	∞	9	∞	0	5	∞	10	∞	H	0	M	0	0	P	0	T	0
P	7	17	4	5	0	∞	∞	3	P	D	D	U	H	0	0	0	R
S	∞	2	9	∞	∞	0	8	∞	S	0	M	U	0	0	0	T	0
T	∞	∞	∞	10	∞	8	0	5	T	0	0	0	H	0	S	0	R
R	∞	6	∞	∞	3	∞	5	0	R	0	M	0	0	P	0	T	0

Через вершину M:

	D	M	U	H	P	S	T	R
D	0	10	7	19	7	12	∞	16
M	10	0	7	9	17	2	∞	6
U	7	7	0	16	4	9	∞	13
H	19	9	16	0	5	11	10	15
P	7	17	4	5	0	19	∞	3
S	12	2	9	11	19	0	8	8
T	∞	∞	∞	10	∞	8	0	5
R	16	6	13	15	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	M	P	M	0	M
M	D	0	U	H	D	S	0	R
U	D	M	0	M	P	S	0	M
H	M	M	M	0	P	M	T	M
P	D	D	U	H	0	M	0	R
S	M	M	U	M	M	0	T	M
T	0	0	0	H	0	S	0	R
R	M	M	M	M	P	M	T	0

Через вершину U:

	D	M	U	H	P	S	T	R
D	0	10	7	19	7	12	∞	16
M	10	0	7	9	11	2	∞	6
U	7	7	0	16	4	9	∞	13
H	19	9	16	0	5	11	10	15
P	7	11	4	5	0	13	∞	3
S	12	2	9	11	13	0	8	8
T	∞	∞	∞	10	∞	8	0	5
R	16	6	13	15	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	M	P	M	0	M
M	D	0	U	H	U	S	0	R
U	D	M	0	M	P	S	0	M
H	M	M	M	0	P	M	T	M
P	D	U	U	H	0	U	0	R
S	M	M	U	M	U	0	T	M
T	0	0	0	H	0	S	0	R
R	M	M	M	M	P	M	T	0

Через вершину H:

	D	M	U	H	P	S	T	R
D	0	10	7	19	7	12	29	16
M	10	0	7	9	11	2	19	6
U	7	7	0	16	4	9	26	13
H	19	9	16	0	5	11	10	15
P	7	11	4	5	0	13	15	3
S	12	2	9	11	13	0	8	8
T	29	19	26	10	15	8	0	5
R	16	6	13	15	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	M	P	M	M	M
M	D	0	U	H	U	S	H	R
U	D	M	0	M	P	S	M	M
H	M	M	M	0	P	M	T	M
P	D	U	U	H	0	U	H	R
S	M	M	U	M	U	0	T	M
T	H	H	H	H	H	S	0	R
R	M	M	M	M	P	M	T	0

Через вершину P:

	D	M	U	H	P	S	T	R
D	0	10	7	12	7	12	22	10
M	10	0	7	9	11	2	19	6
U	7	7	0	9	4	9	19	7
H	12	9	9	0	5	11	10	8
P	7	11	4	5	0	13	15	3
S	12	2	9	11	13	0	8	8
T	22	19	19	10	15	8	0	5
R	10	6	7	8	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	P	P	M	P	P
M	D	0	U	H	U	S	H	R
U	D	M	0	P	P	S	P	P
H	P	M	P	0	P	M	T	P
P	D	U	U	H	0	U	H	R
S	M	M	U	M	U	M	T	M
T	H	H	H	H	H	S	0	R
R	P	M	P	P	P	M	T	0

Через вершину S:

	D	M	U	H	P	S	T	R
D	0	10	7	12	7	12	20	10
M	10	0	7	9	11	2	10	6
U	7	7	0	9	4	9	17	7
H	12	9	9	0	5	11	10	8
P	7	11	4	5	0	13	15	3
S	12	2	9	11	13	0	8	8
T	20	10	17	10	15	8	0	5
R	10	6	7	8	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	P	P	M	M	P
M	D	0	U	H	U	S	S	R
U	D	M	0	P	P	S	S	P
H	P	M	P	0	P	M	T	P
P	D	U	U	H	0	U	H	R
S	M	M	U	M	U	M	T	M
T	S	S	S	H	H	S	0	R
R	P	M	P	P	P	M	T	0

Через вершину T: - изменений нет

	D	M	U	H	P	S	T	R
D	0	10	7	12	7	12	20	10
M	10	0	7	9	11	2	10	6
U	7	7	0	9	4	9	17	7
H	12	9	9	0	5	11	10	8
P	7	11	4	5	0	13	15	3
S	12	2	9	11	13	0	8	8
T	20	10	17	10	15	8	0	5
R	10	6	7	8	3	8	5	0

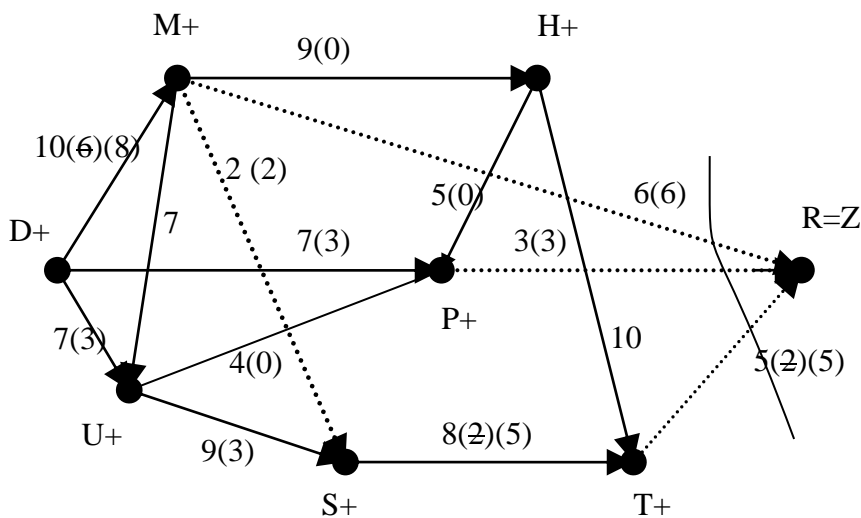
	D	M	U	H	P	S	T	R
D	0	M	U	P	P	M	M	P
M	D	0	U	H	U	S	S	R
U	D	M	0	P	P	S	S	P
H	P	M	P	0	P	M	T	P
P	D	U	U	H	0	U	H	R
S	M	M	U	M	U	M	T	M
T	S	S	S	H	H	S	0	R
R	P	M	P	P	P	M	T	0

Через вершину R Результирующие таблицы.

	D	M	U	H	P	S	T	R
D	0	10	7	12	7	12	15	10
M	10	0	7	9	9	2	10	6
U	7	7	0	9	4	9	12	7
H	12	9	9	0	5	11	10	8
P	7	9	4	5	0	11	8	3
S	12	2	9	11	11	0	8	8
T	15	10	12	10	8	8	0	5
R	10	6	7	8	3	8	5	0

	D	M	U	H	P	S	T	R
D	0	M	U	P	P	M	P	P
M	D	0	U	H	R	S	S	R
U	D	M	0	P	P	S	P	P
H	P	M	P	0	P	M	T	P
P	D	R	U	H	0	R	R	R
S	M	M	U	M	M	0	T	M
T	R	S	R	H	R	S	0	R
R	P	M	P	P	P	M	T	0

Поиск максимального потока в сети (метод Форда-Фалкерсона).



Этап насыщения:

$$D \xrightarrow{10} M \xrightarrow{6} R \quad \min=6$$

$$D \xrightarrow{7} P \xrightarrow{3} R \quad \min=3$$

$$D \xrightarrow{4} M \xrightarrow{2} S \xrightarrow{8} T \xrightarrow{5} R \quad \min=2$$

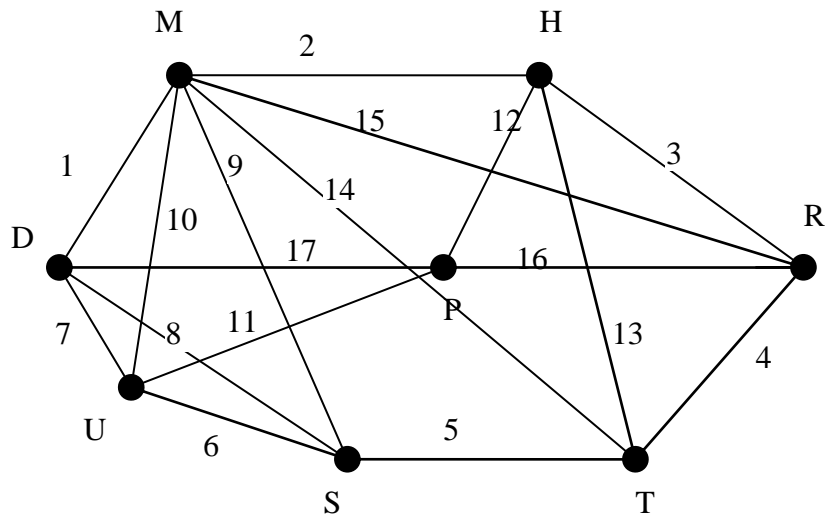
$$D \xrightarrow{7} U \xrightarrow{9} S \xrightarrow{6} T \xrightarrow{3} R \quad \min=3;$$

Максимальный поток сети: $\Phi=6+3+5+8+3+3=14$.

По методу Форда-Фалкерсона построено минимальное сечение, разделяющее помеченные вершины от непомеченных. Максимальный поток по методу Форда-Фалкерсона: $6+3+5=14$.

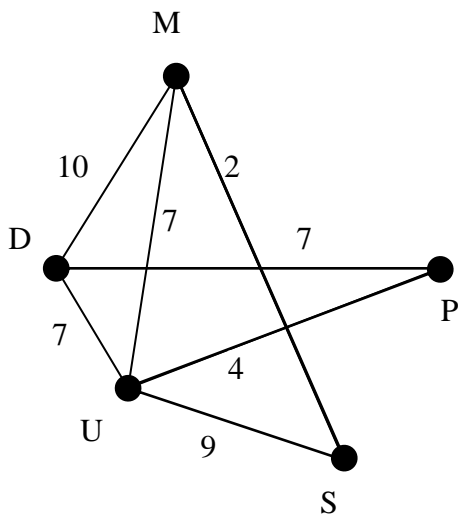
Эйлерова цепь и цикл.

В данном графе 6 вершин с нечетными степенями, поэтому добавим ребра DS, MT, HR. На рисунке ребра пронумерованы в порядке их обхода.

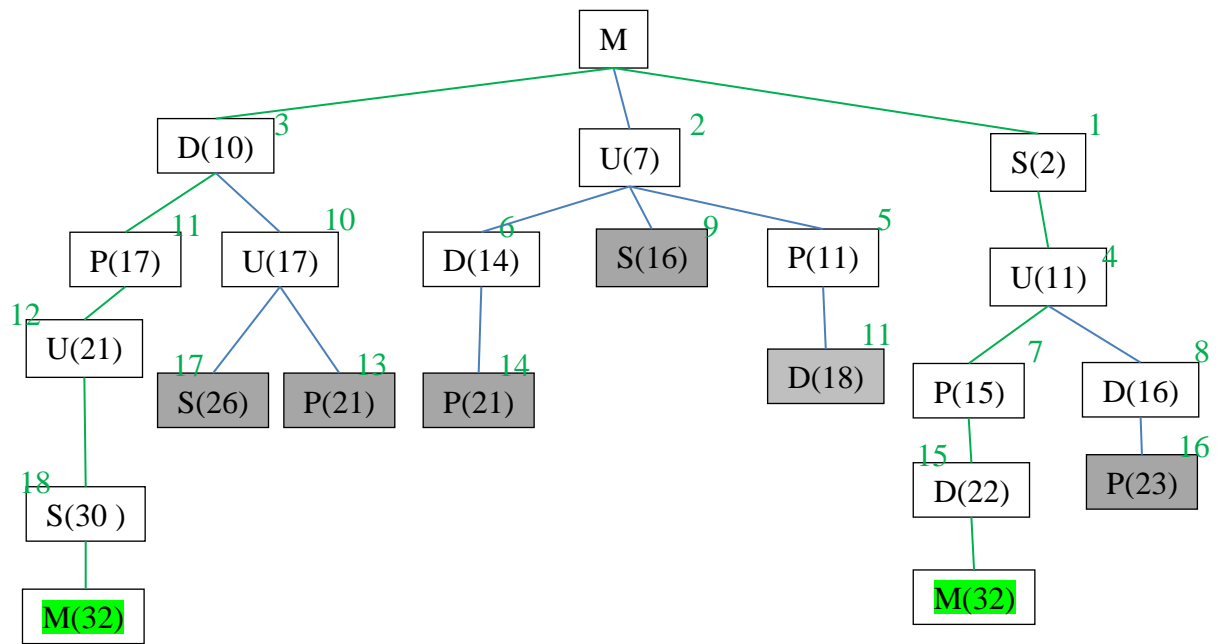


Задача коммивояжера. Гамильтонова цепь и цикл.

Оставим в графе 5 вершин:



Усеченное граф-дерево (метод ветвей и границ)



Зеленые цифры показывают порядок выбора вершины с минимальной оценкой.

Желтый фон вершины – отсеченные вершины.

Серый фон вершины – тупиковые вершины.

Зеленый цветом показан самый короткий гаммильтонов цикл (ответ задачи коммивояжера) - 32.

ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА

1. Рязанов, Ю. Д. Дискретная математика : учебное пособие / Ю. Д. Рязанов ; под редакцией В. И. Пустовая. — 2-е изд. — Белгород : Белгородский государственный технологический университет им. В.Г. Шухова, ЭБС АСВ, 2016. — 298 с. — ISBN 978-5-361-00364-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/80509.html>
2. Пашуева, И. М. Дискретная математика в информационных системах и технологиях : учебное пособие / И. М. Пашуева, А. Н. Шелковой, Н. А. Ююкин. — Воронеж : Воронежский государственный технический университет, ЭБС АСВ, 2018. — 183 с. — ISBN 978-5-7731-0718-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/93256.html>
3. Прокопенко, Н. Ю. Дискретная математика : учебное пособие / Н. Ю. Прокопенко. — Нижний Новгород : Нижегородский государственный архитектурно-строительный университет, ЭБС АСВ, 2016. — 252 с. — ISBN 978-5-528-00127-2. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/80893.html>
4. Рогова, Н. В. Дискретная математика : учебное пособие / Н. В. Рогова. — Самара : Поволжский государственный университет телекоммуникаций и информатики, 2017. — 143 с. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <https://www.iprbookshop.ru/75372.html>

МЕТОДИЧЕСКИЕ УКАЗАНИЯ
к выполнению индивидуального задания по дисциплине
«Дискретная математика»

Составитель:

Чередникова Ольга Юрьевна – кандидат технических наук, доцент кафедры компьютерной инженерии ГОУВПО «ДОННТУ»

Ответственный за выпуск:

Аноприенко Александр Яковлевич – кандидат технических наук, доцент, заведующий кафедрой «Компьютерная инженерия» ГОУВПО «ДОННТУ»