ГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«ДОНЕЦКИЙ НАЦИОНАЛЬНЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

К РАСЧЕТНОЙ РАБОТЕ
ПО ДИСЦИПЛИНЕ
"ДИСКРЕТНАЯ МАТЕМАТИКА"

для студентов очной и заочной формы обучения направления подготовки
09.03.01 «Информатика и вычислительная техника»
профилей
«Вычислительные машины, комплексы, системы и сети»
«Программное обеспечение средств вычислительной техники»

Рассмотрено
на заседании кафедры КИ
протокол № 5 от 30.01.2017

Утверждено на заседании учебно – издательского совета ДонНТУ

Протокол № 2 от 23.03. 2017 г.

Донецк 2017

УДК 519.17

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К РАСЧЕТНОЙ РАБОТЕ ПО ДИСЦИПЛИНЕ «ДИСКРЕТНАЯ МАТЕМАТИКА» (для студентов очной, заочной формы обучения) направления подготовки 09.03.01 «Информатика и вычислительная техника» профилей «Вычислительные машины, комплексы, системы и сети», «Программное обеспечение средств вычислительной техники» / Составитель: Чередникова О.Ю. - Донецк: ДонНТУ, 2017. – 31 с.

Целью расчетной работы по дисциплине “Дискретная математика” является углубление знаний студентов одного из разделов дискретной математики – теории графов.

Составитель: О.Ю.Чередникова, к.т.н., доцент

Рецензенты: В.А. Краснокутский, к.т.н., доц.
Т.А. Васяева, к.т.н., доц.

Оглавление

ЗАДАНИЯ К РАСЧЕТНОЙ РАБОТЕ

Выполнить ***ручной расчет*** для перечисленных ниже задач применительно к графу в соответствии с вариантом:

- Способы хранения информации о графе при программировании (матрицы смежности, инцидентности, список пар, список смежности);
- Определение численных метрических характеристик:
  - радиус
  - диаметр
  - хроматическое число
  - хроматический класс
  - цикломатическое число (определенное по формуле и с помощью построения остова);
- Матрица достижимости (для ориентированного графа)
- Минимальные расстояния между всеми вершинами графа – ручной проход алгоритма Флойда;
- Максимальный поток в сети;
- Эйлеров путь или цикл;
- Задача коммивояжера (Гаммильтонов путь или цикл).

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

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 <X,Y> соответствует дуге <X,Y>, 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 r, содержащей вершину r.строка и указатель r.след на следующую запись в списке (r.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.

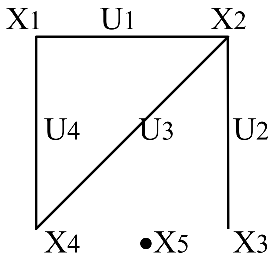


Рис.1 - Граф

матрица смежности

	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

n (строки), n (столбцы)

вершины

матрица инцидентности

	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

n, m

ребра

список пар

ребро	нач	конец
U1	1	2
U2	2	3
U3	2	4
U4	1	4

список смежности

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 $p(x_i, x_j) = \min\limits_k l(M_{ij}^k)$.

Если xi и xj не соединены цепью (т.е. принадлежат разным компонентам), то P(xi,xj) =∞ .

Наибольшее расстояние между любыми двумя вершинами графа - его **<u>диаметр</u>**:

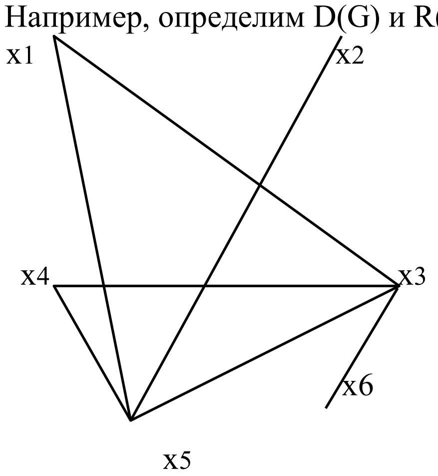
$$\alpha(G) = \max_{\forall i,j} P(x_i, x_j) = \max_{\forall i,j} \min_k l(M_{i,j}^k);$$

<u>Радиус</u> <u>вершины</u> x_i: $r(x_i) = \max\limits_{\forall j} P(x_i, x_j)$;

<u>Радиус графа</u>- минимальный из радиусов вершин: $r(G) = \min\limits_{\forall i} r(x_i) = \min\limits_{\forall i} \max\limits_{\forall j} P(x_i, x_j)$;

Для более наглядного поиска радиуса и диметра 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):



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 – 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. Граф G называется p-хроматическим, если он допускает правильную вершинную окраску в p цветов. Наименьшее число цветов p которыми можно раскрасить вершины графа, называется **хроматическим числом** графа.

Например (рис.4):

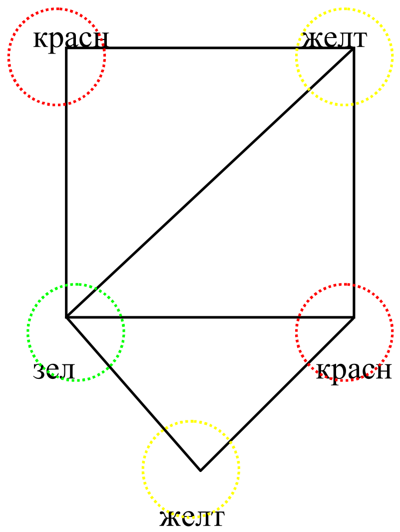


Рис.4 – Раскраска графа

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) ≥ 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.

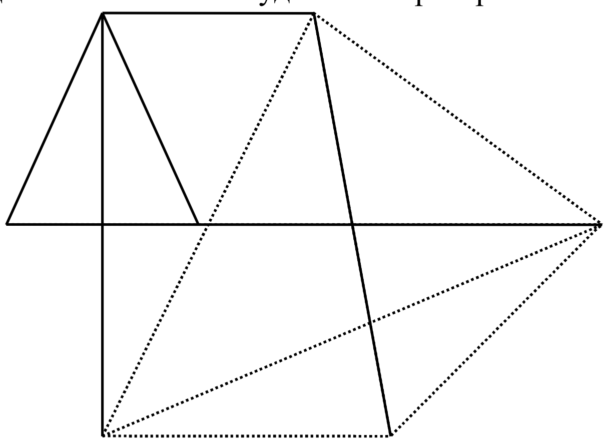


Рис. 5 – Остов графа

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: $P_{ij} = \begin{cases} 1, if \exists x_i \to x_j \\ 0, else \end{cases}$

Минимальные расстояния между всеми вершинами графа

В РГР 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.

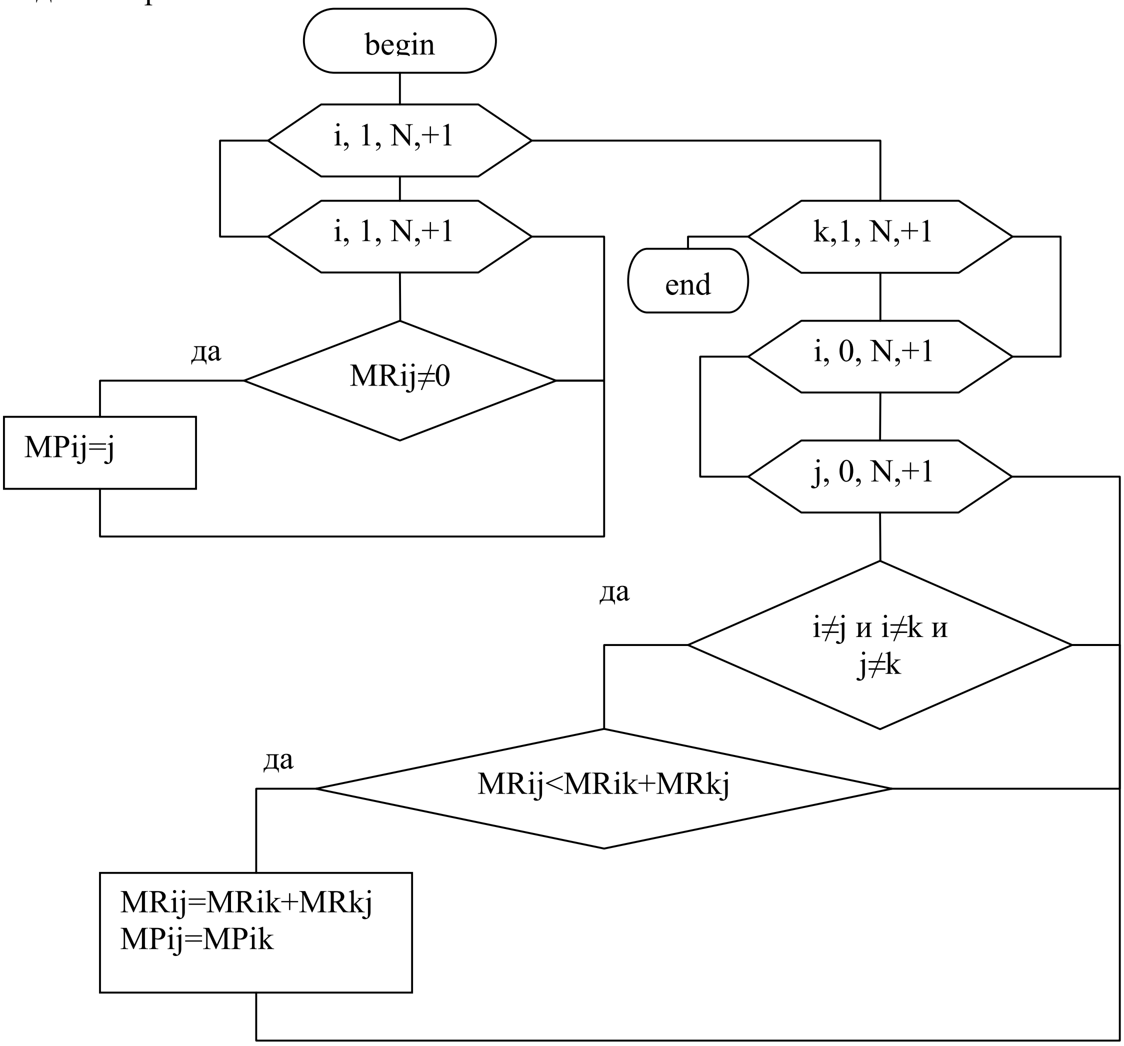


Рис.6 – Блок-схема алгоритма Флойда

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) $0 \le F(U) \le C(U)$

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 Эйлерова 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 завершиться 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 Эйлерова цикла не соблюдены, то в РГР это аргументированно указывается.

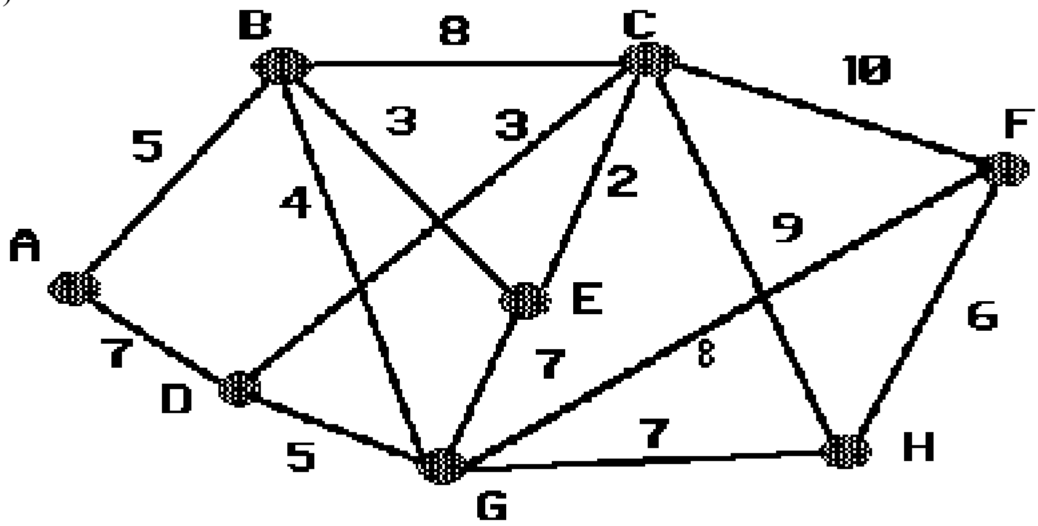
Задача коммивояжера (Гаммильтонов путь или цикл)

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 гаммильтонового 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:

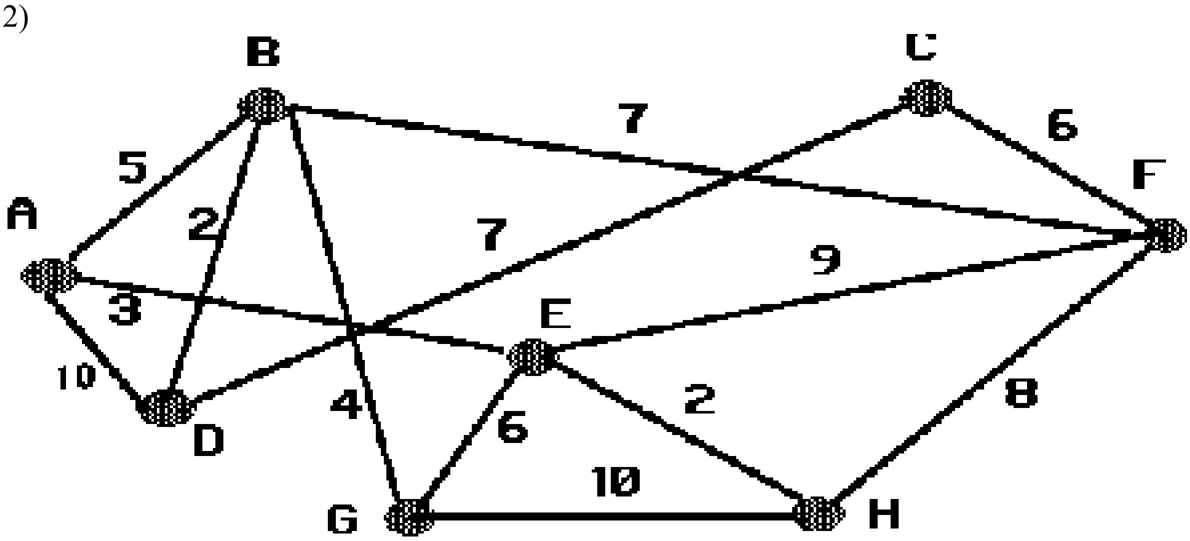
1. Принимает какую-либо вершину за начальную (это корень дерева).
2. Добавляем в дерево вершины, смежные с выбранной и соединяем их ребром с выбранной вершиной. При этом учитываем, что выбирать вершины, через которые мы уже проходили в этой ветви мы не можем. Если все вершины графа были пройдены в ветви, то возвращаемся в начальную вершину – это гаммильтонов 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.

ВАРИАНТЫ ВЫБОРА ГРАФА

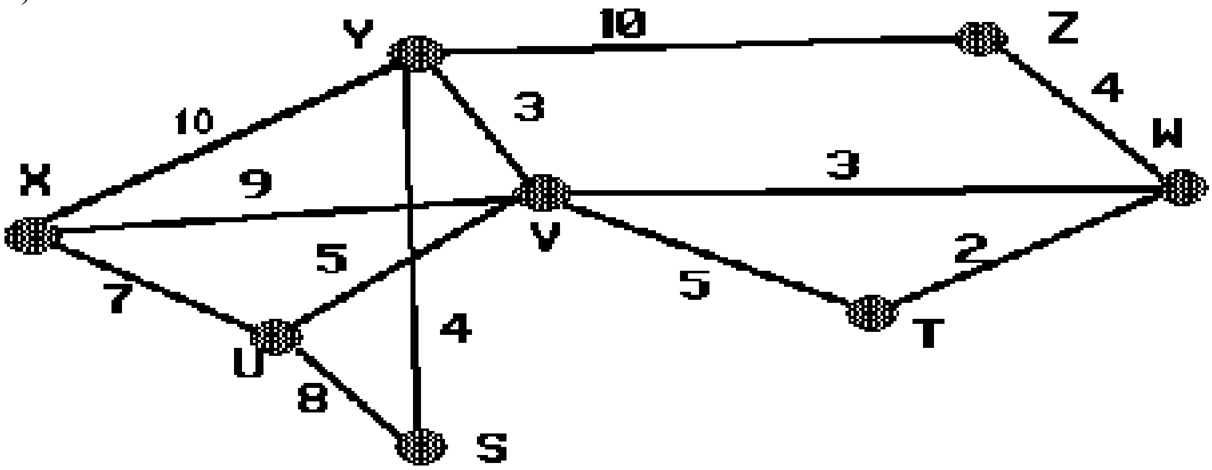
1)



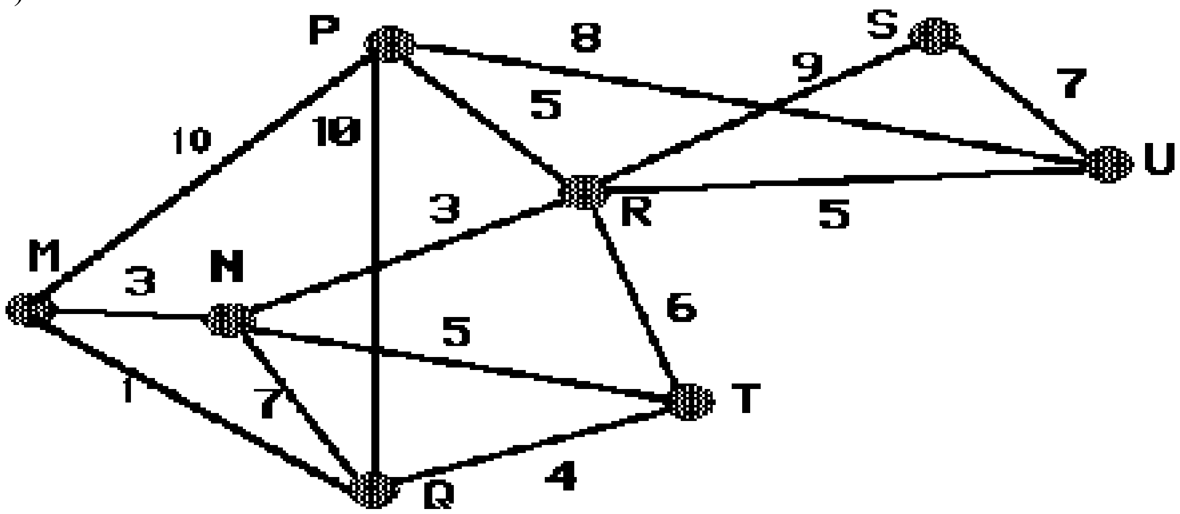
2)



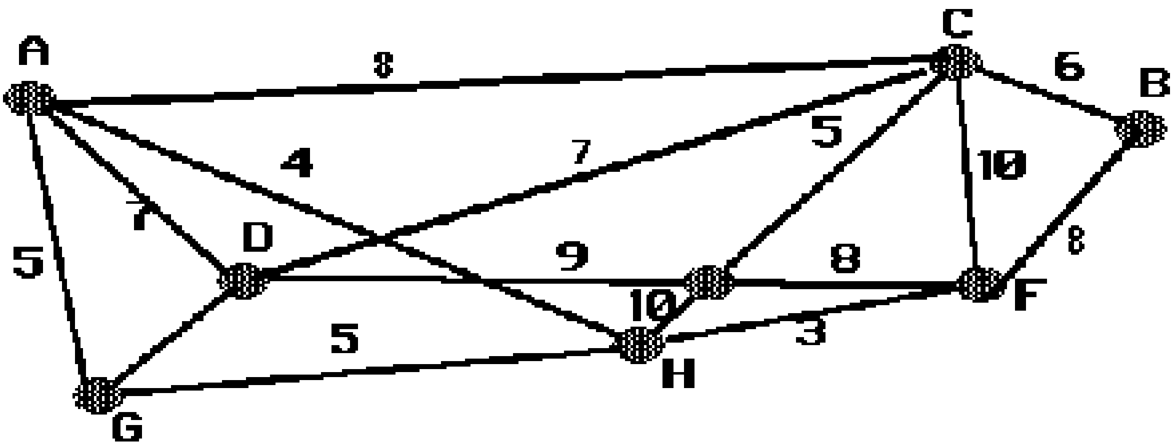
3)



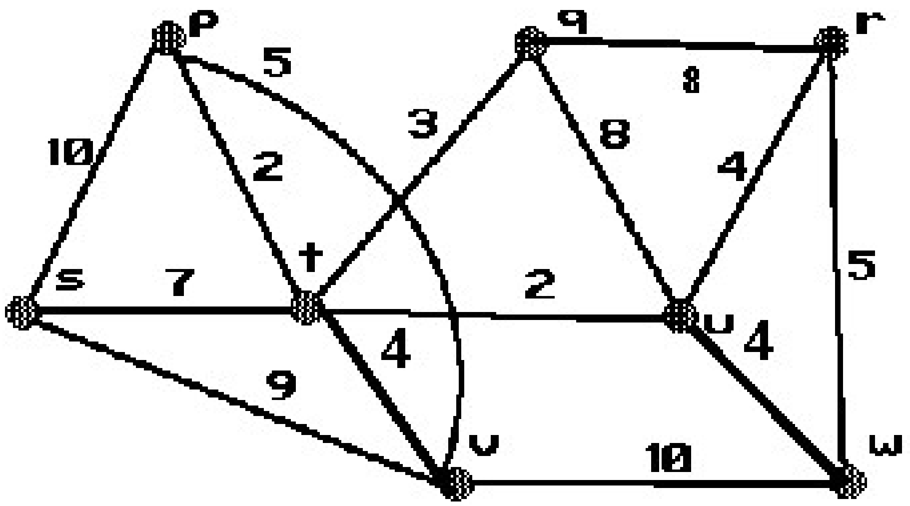
4)


P
S
U
R
M
N
T
Q
8
9
7
10
10
5
3
5
3
6
5
1
7
4

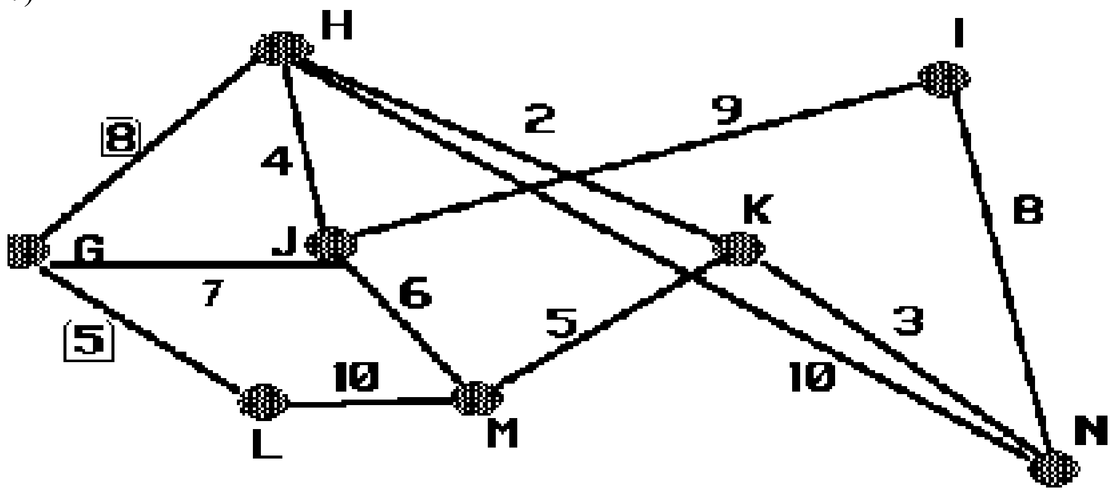
5)


A
C
B
D
F
H
G
8
6
5
7
4
10
7
5
9
8
8
10
5
3

6)

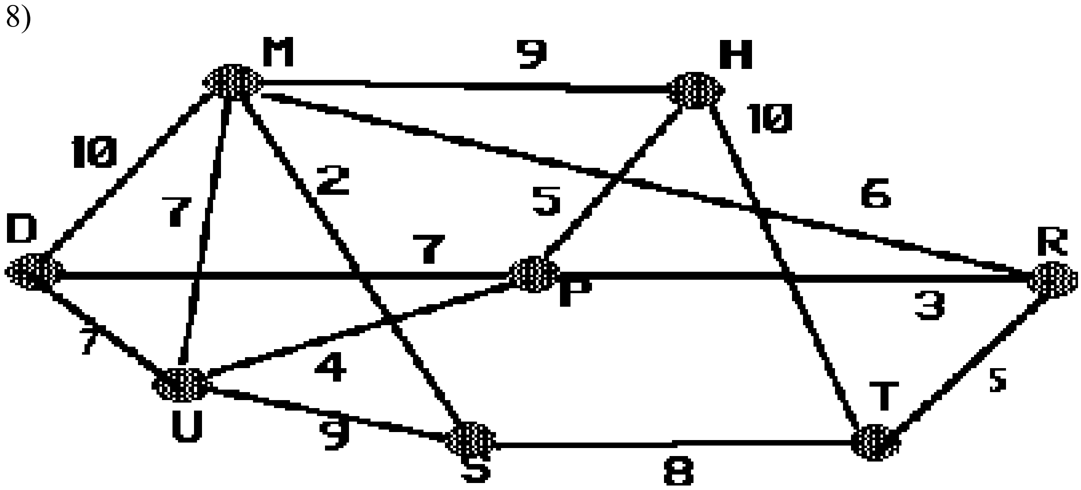

p
q
r
s
t
u
v
w
5
8
10
3
2
8
4
7
2
5
4
4
9
10

7)



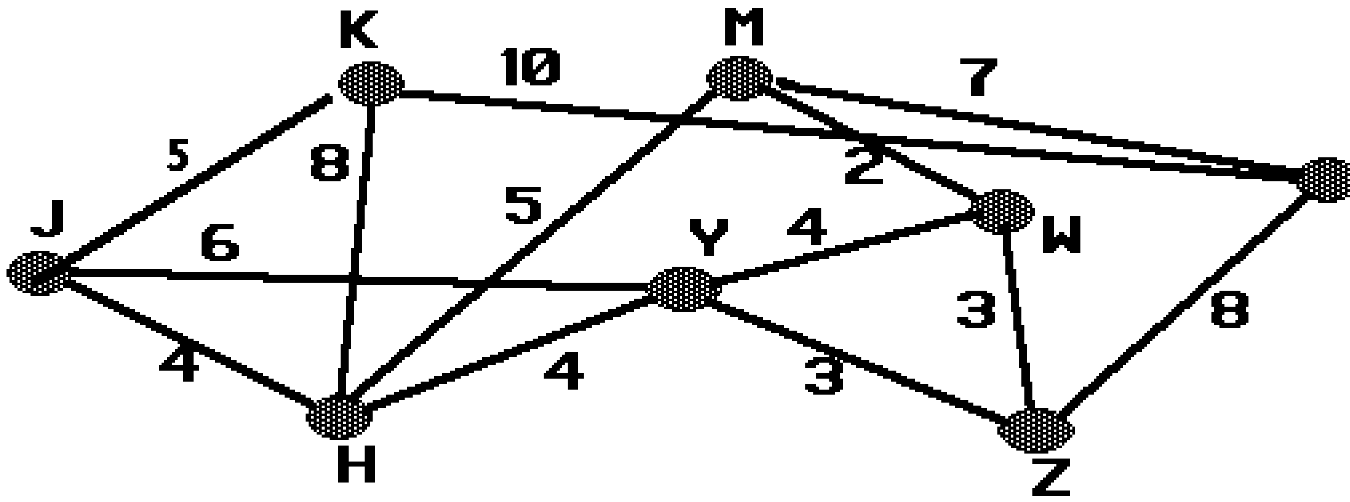
H
I
8
4
2
9
8
G
J
K
7
6
5
3
5
10
10
N
L
M

8)



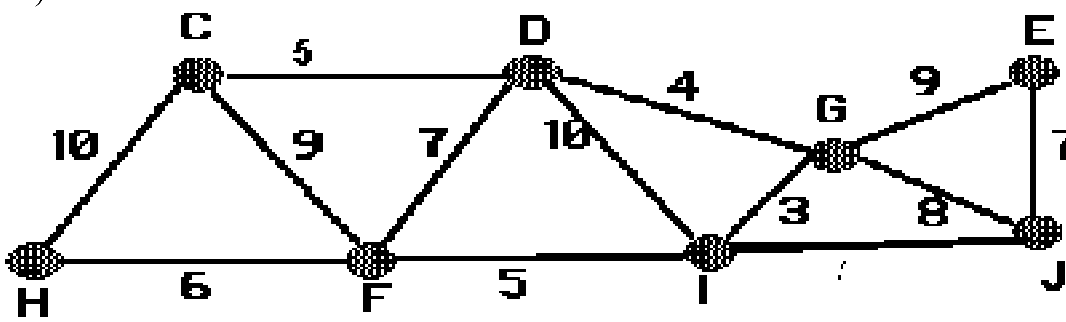
M
9
H
10
10
2
6
7
5
D
7
R
P
3
7
4
5
T
U
9
S
8

9)

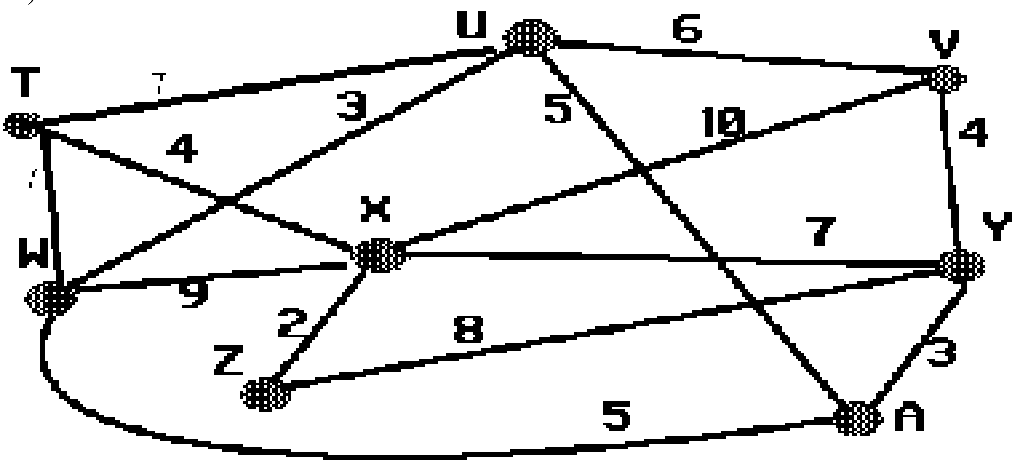


K
10
M
7
5
8
2
5
J
6
Y
4
W
3
8
4
4
3
H
Z

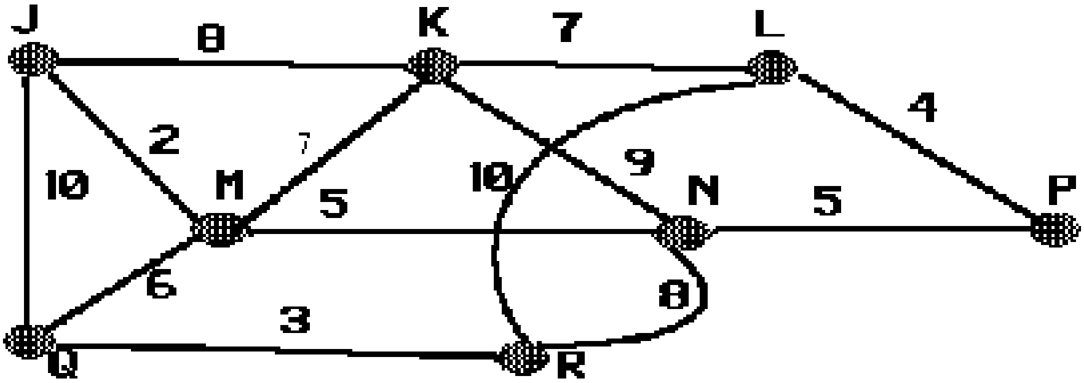
10)


C
D
E
5
4
9
G
10
9
7
10
7
3
8
H
6
F
5
I
J

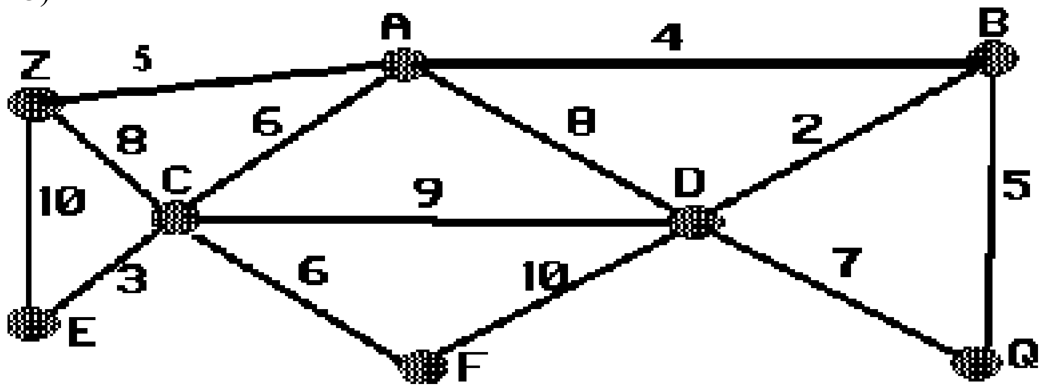
11)


U
6
V
T
3
5
10
4
4
X
7
Y
W
9
2
8
Z
3
5
A

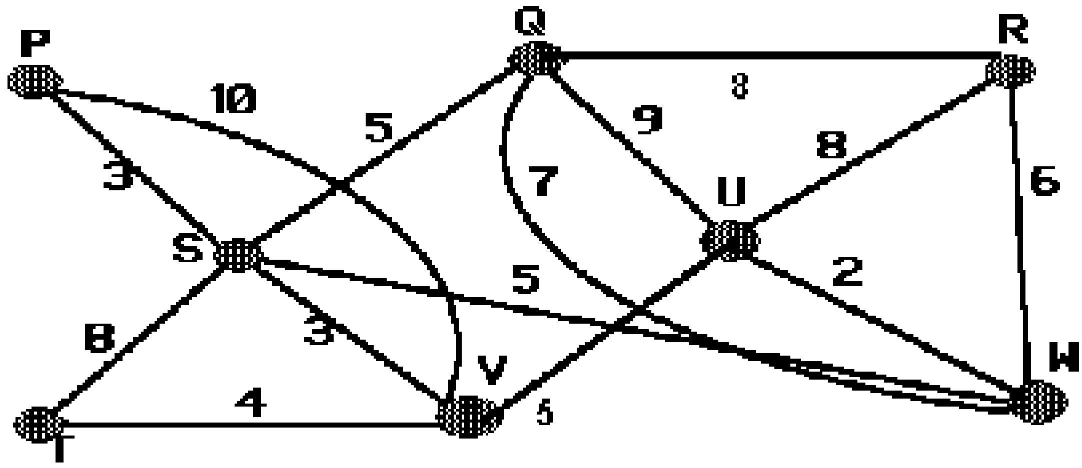
12)


J
8
K
7
L
4
2
10
10
9
M
5
N
5
P
6
8
3
Q
R

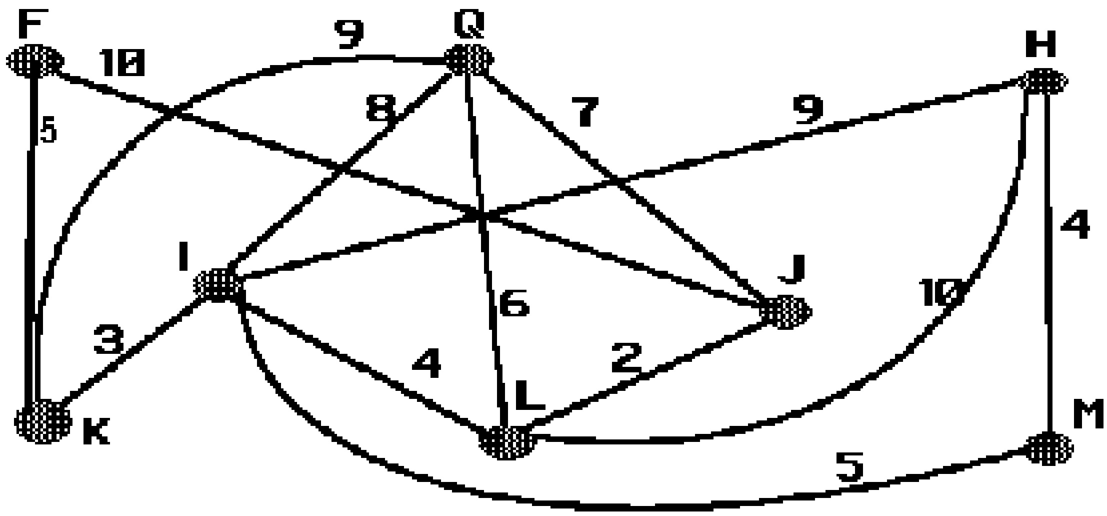
13)


A
B
Z
4
5
8
6
8
2
10
C
D
5
9
3
6
10
7
E
F
Q

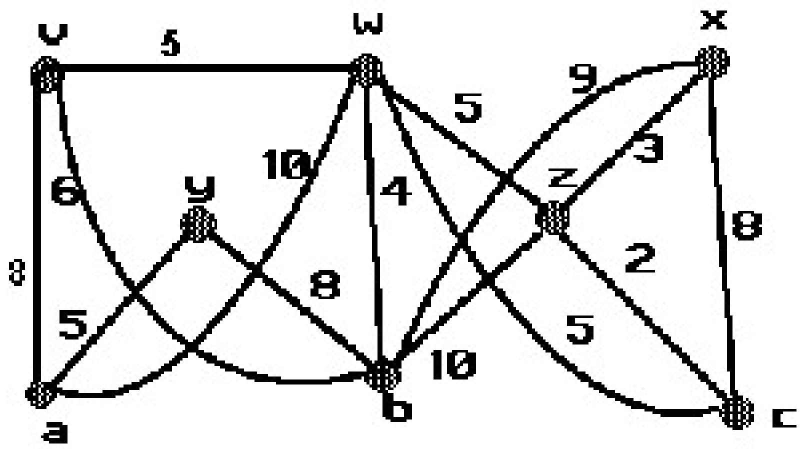
14)


Q
R
P
8
10
5
9
3
8
6
7
U
S
5
2
8
3
V
W
4
5
T

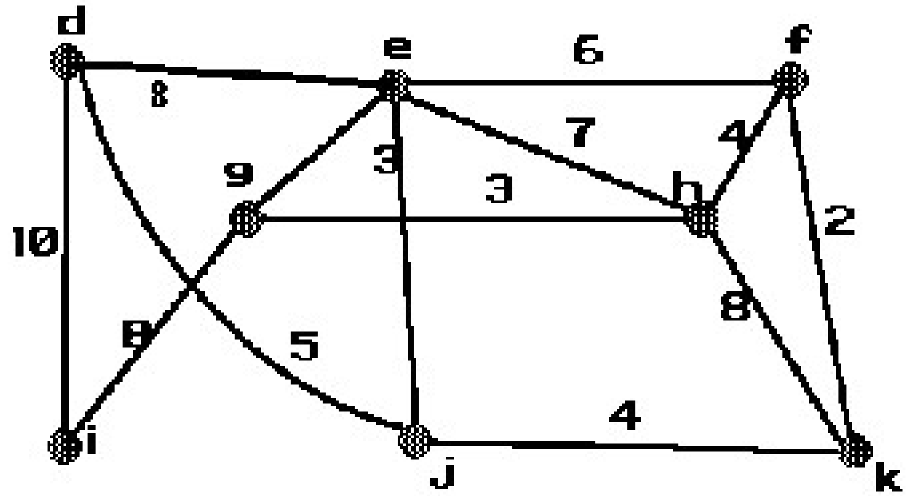
15)


F
Q
H
10
9
8
7
9
5
4
I
J
6
10
3
4
2
L
K
M
5

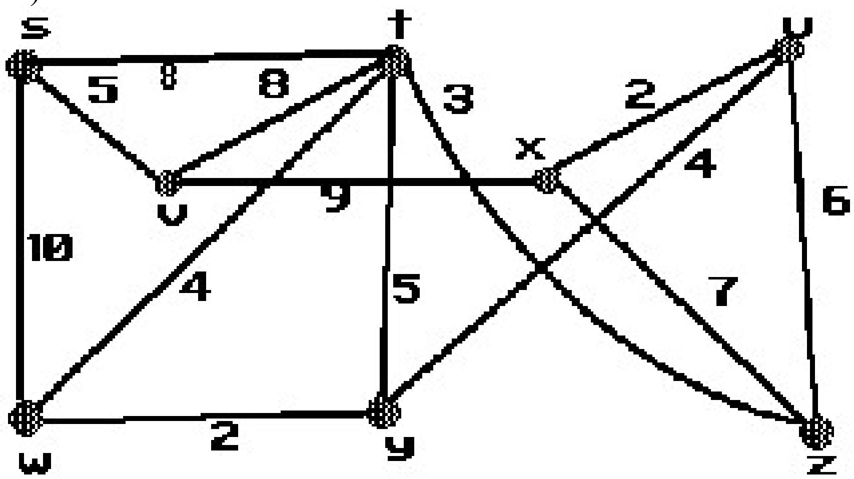
16)


v
w
x
y
z
a
b
c
5
5
9
10
3
6
4
8
8
8
2
5
5
10

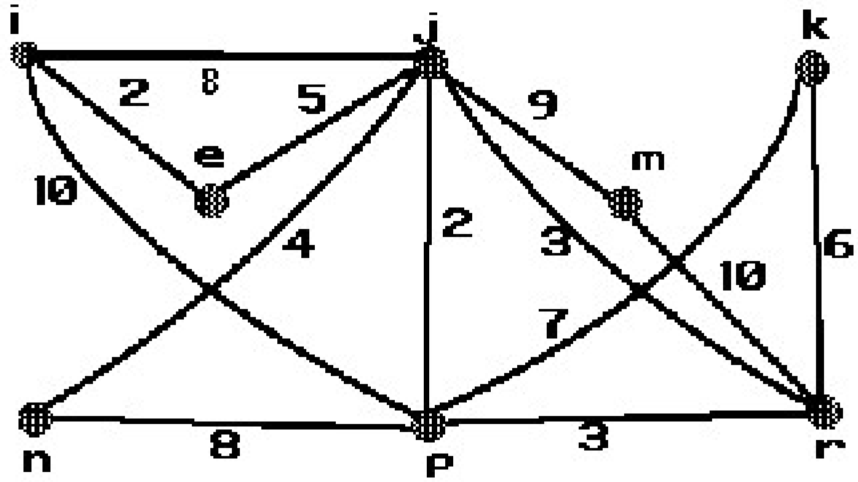
17)


d
e
f
g
h
i
j
k
8
6
7
4
3
3
10
2
8
8
5
4

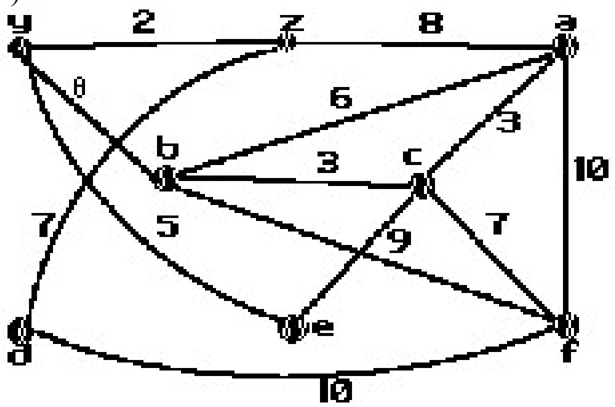
18)


s
t
u
v
x
w
y
z
5
8
8
3
2
4
9
6
10
4
5
7
2

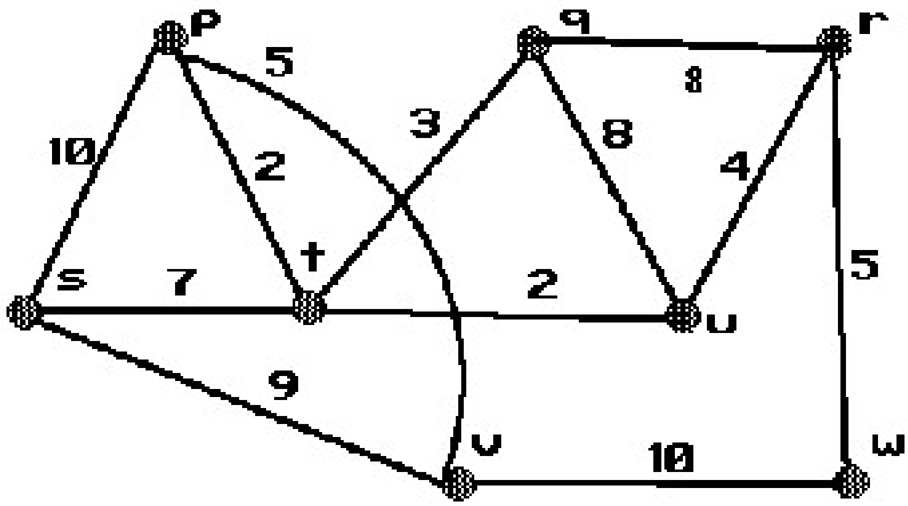
19)


i
j
k
2
8
5
9
e
m
10
2
3
6
4
10
7
8
3
n
p
r

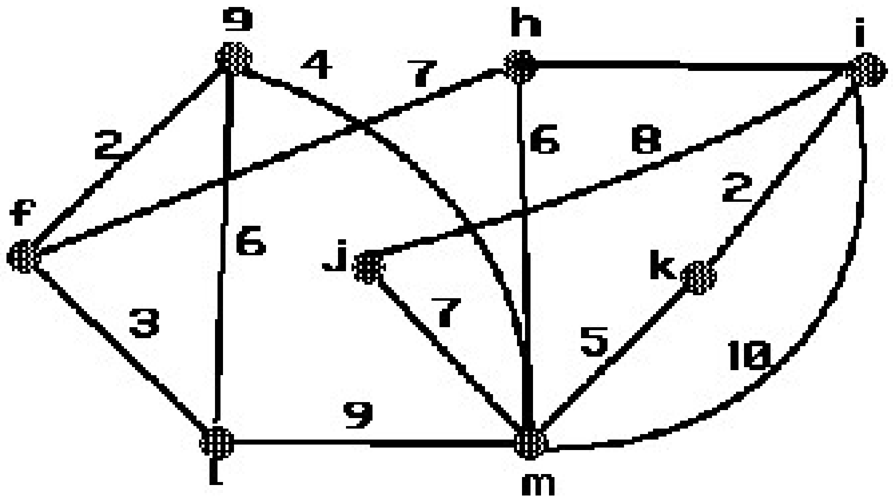
20)


y
2
z
8
a
8
6
3
b
3
c
10
7
5
7
9
e
d
f
10

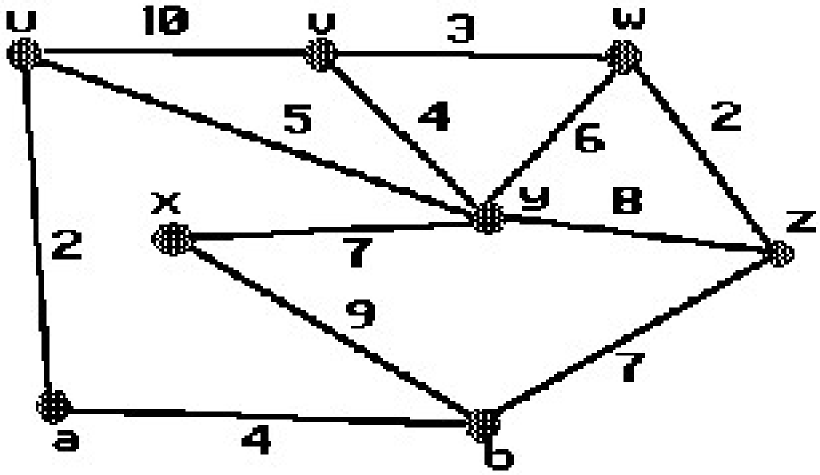
21)


p
5
q
r
8
3
10
8
2
4
5
s
7
t
2
u
9
v
10
w

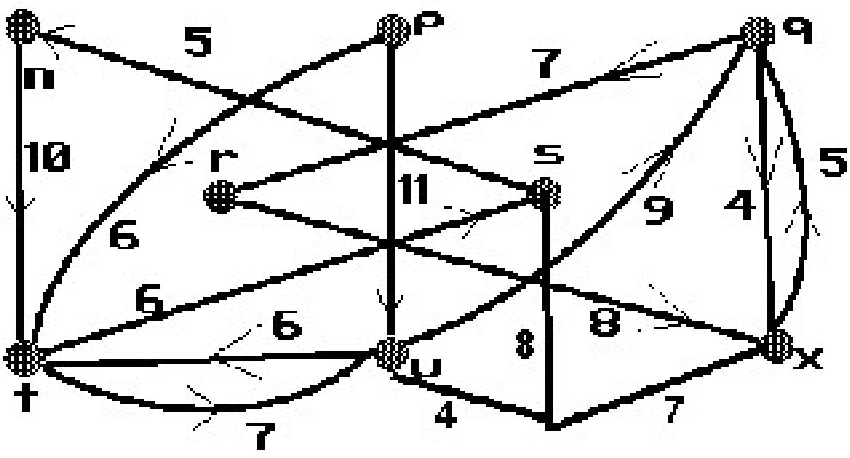
22)


g
h
i
f
j
k
l
m
2
4
7
6
6
8
2
3
7
5
10
9

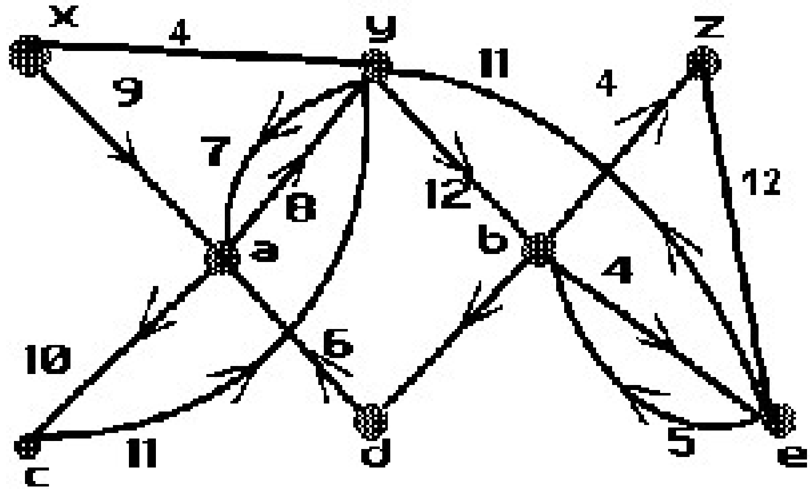
23)


u
v
w
x
y
z
a
b
10
3
5
4
6
2
8
2
7
9
7
4

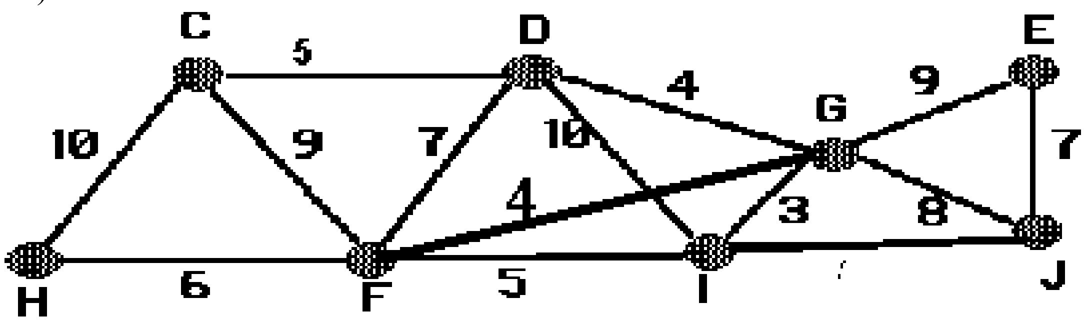
24)


n
p
q
r
s
t
u
x
5
7
10
11
9
4
5
6
6
6
8
8
7
4
7

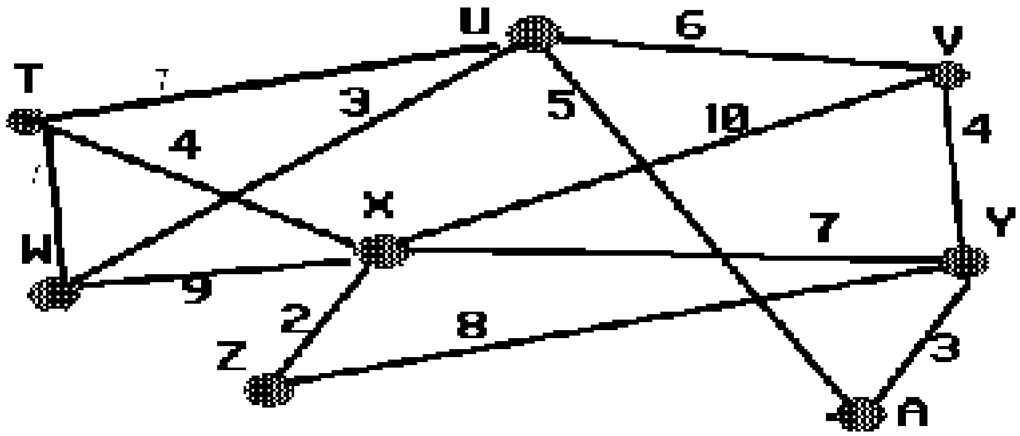
25)


x
4
y
11
z
9
4
7
8
12
12
b
a
4
10
6
11
5
c
d
e

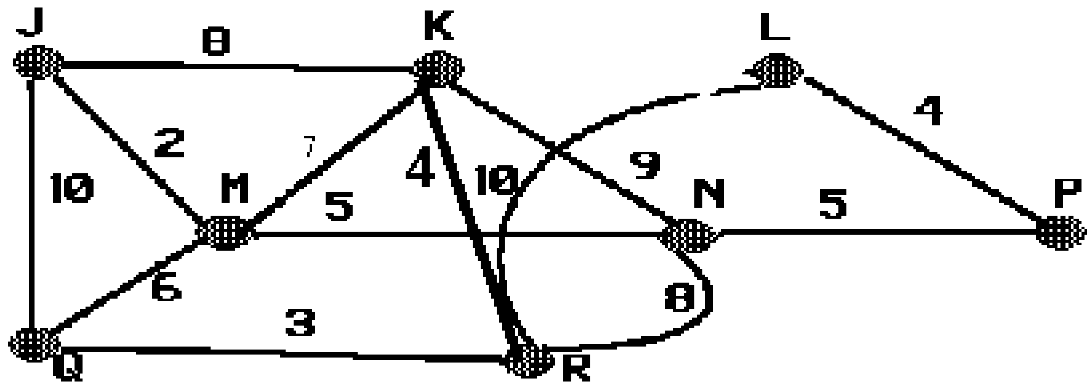
26)


C
5
D
E
4
9
10
9
7
10
G
7
4
3
8
H
6
F
5
I
J

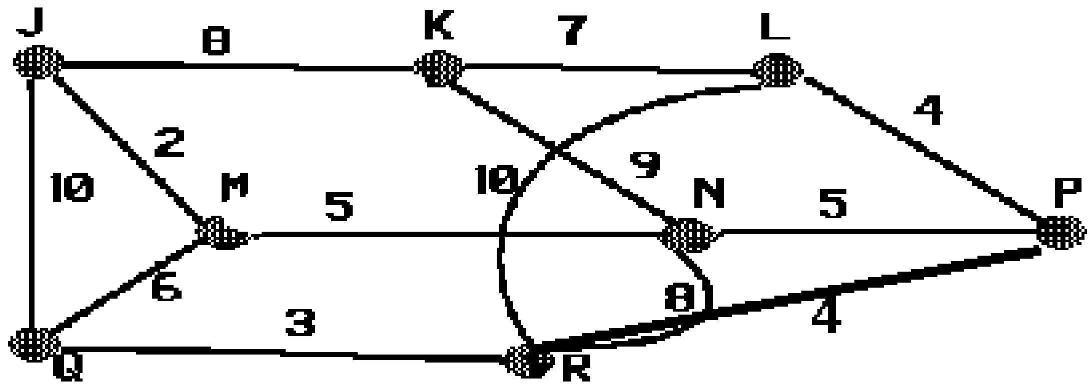
27)


U
6
V
T
3
5
10
4
4
X
W
7
Y
9
2
8
Z
3
A

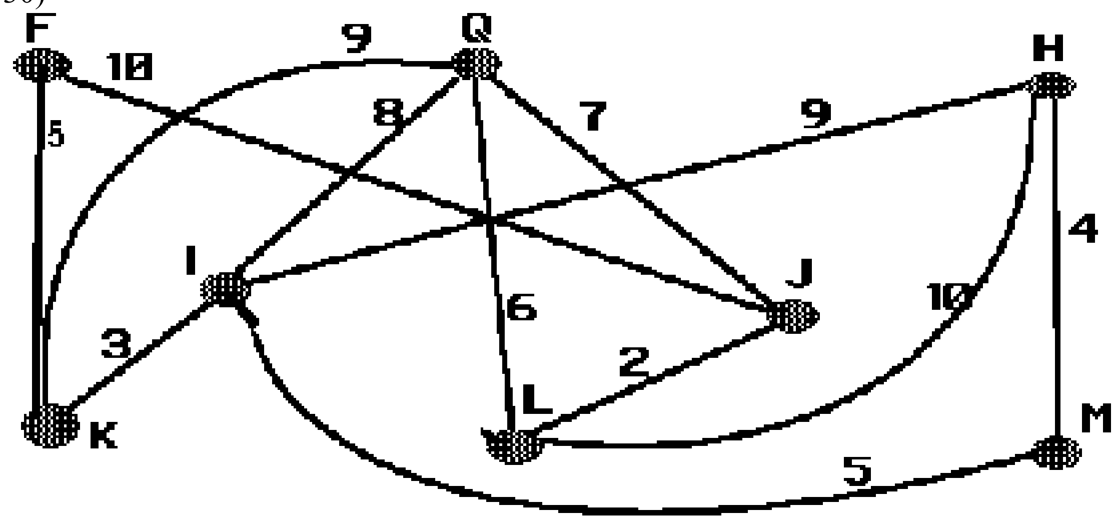
28)


J
8
K
L
2
7
4
10
9
4
10
M
5
N
5
P
6
8
3
Q
R

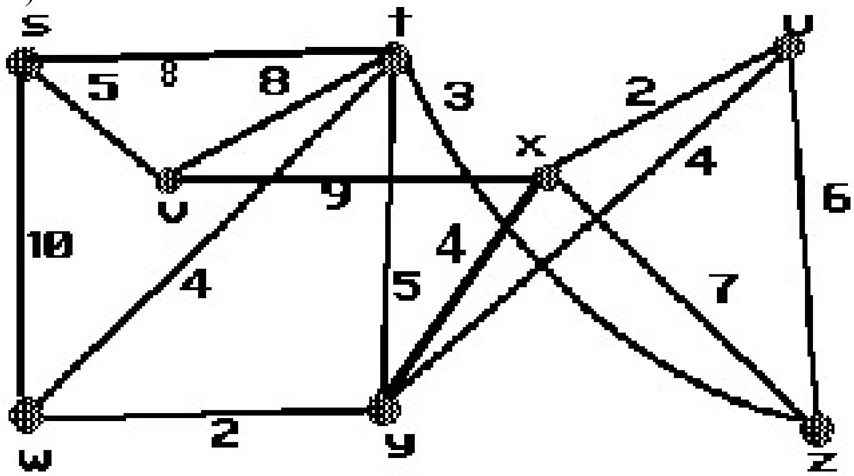
29)


J
8
K
7
L
2
10
9
4
10
M
5
N
5
P
6
8
3
4
Q
R

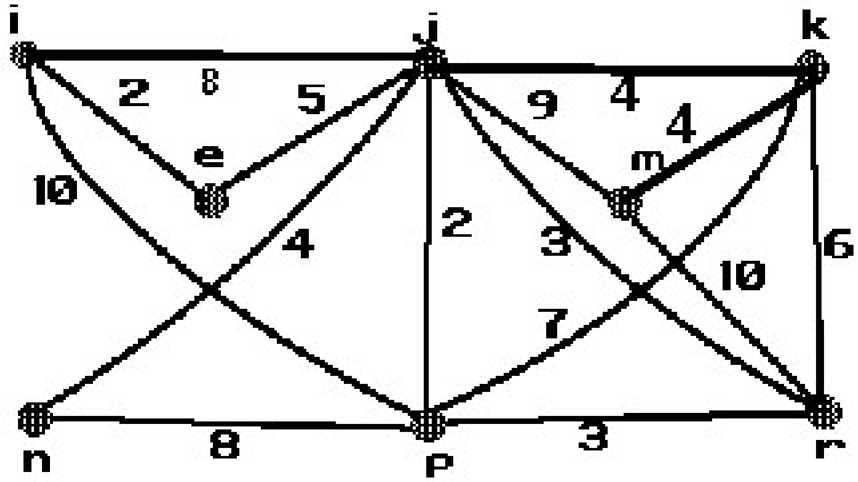
30)


F
10
9
Q
H
5
8
7
9
4
I
J
6
10
3
2
L
K
M
5

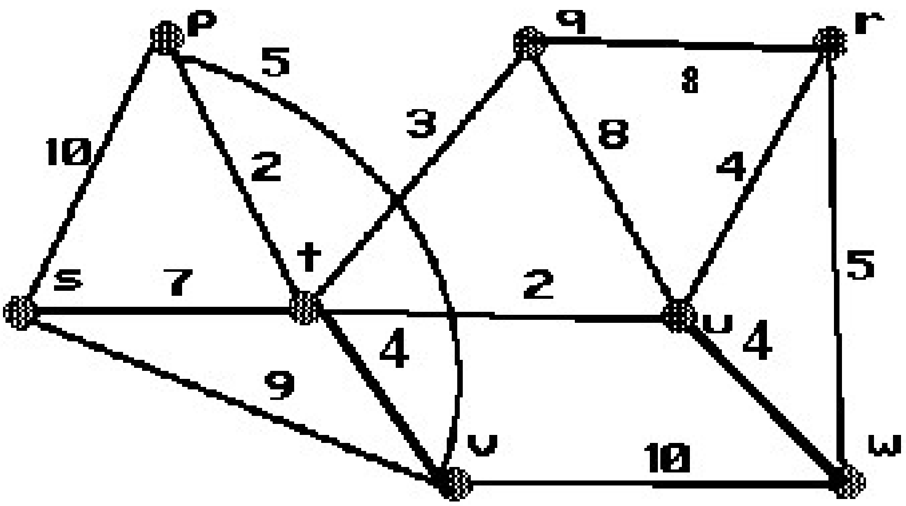
31)


s
t
u
5
8
8
3
2
x
4
v
9
6
10
4
5
4
7
w
2
y
z

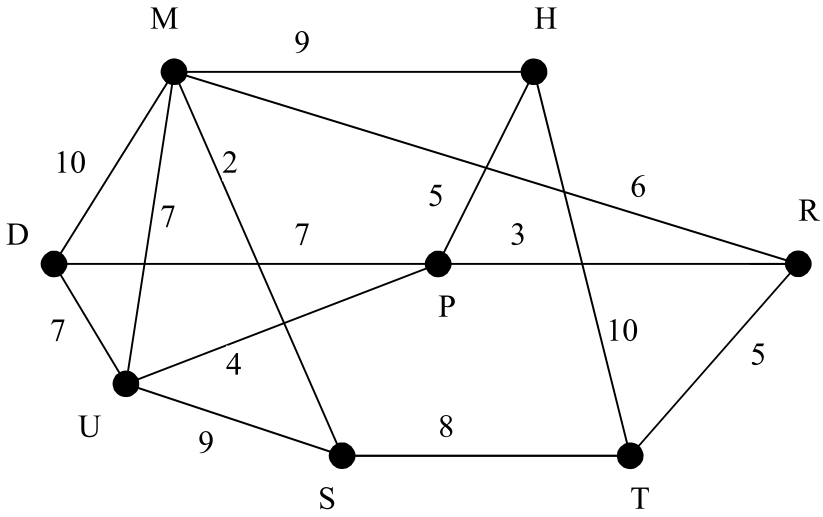
32)


i
j
k
2
8
5
9
4
4
e
m
10
4
2
3
10
6
7
n
8
p
3
r

33)


p
5
q
r
8
10
3
2
8
4
t
5
s
7
2
u
4
4
9
v
10
w

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 смежности:

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

2. Определение метрических характеристик графа:

Количество вершин: 8.

Количество ребер: 14.

Степени вершин:

D	M	U	H	P	S	T	R
3	5	4	3	4	3	3	3

Расстояния между вершинами:

	D	M	U	H	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
H	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

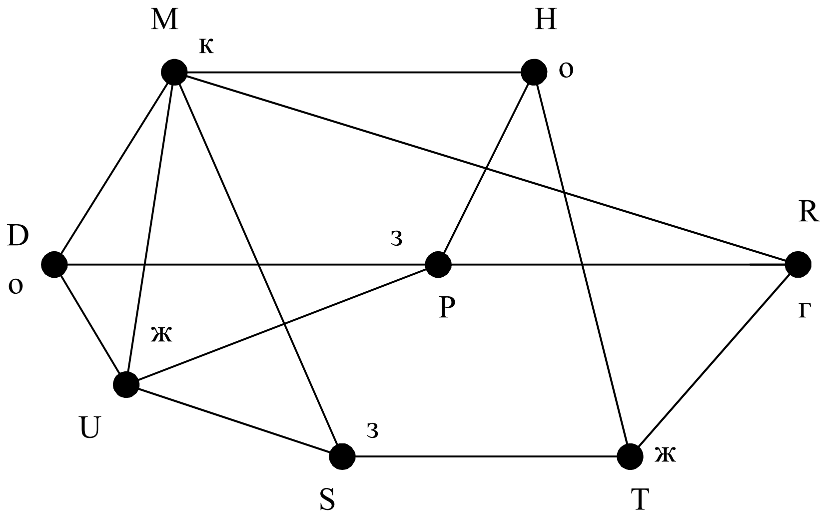
Радиусы вершин:

D	M	U	H	P	S	T	R
3	2	2	2	2	2	3	2

Диаметр графа: 3.

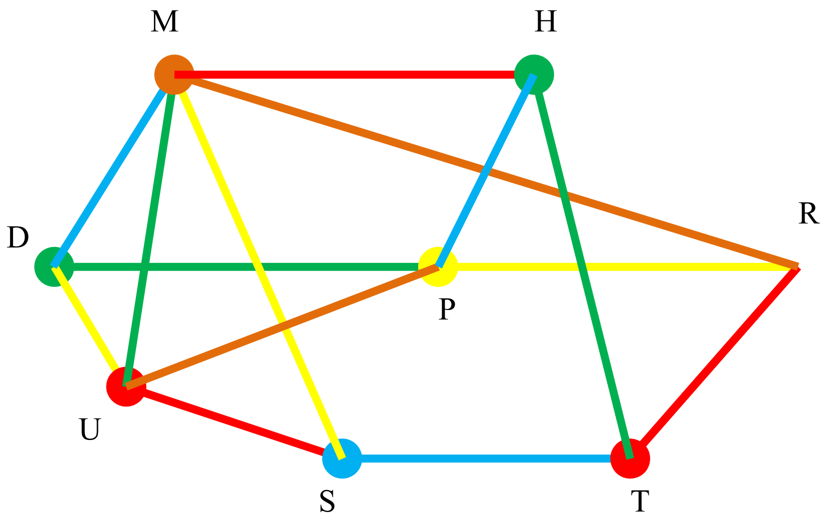
Радиус графа: 2.

Определение хроматического числа графа. Раскраска вершин:



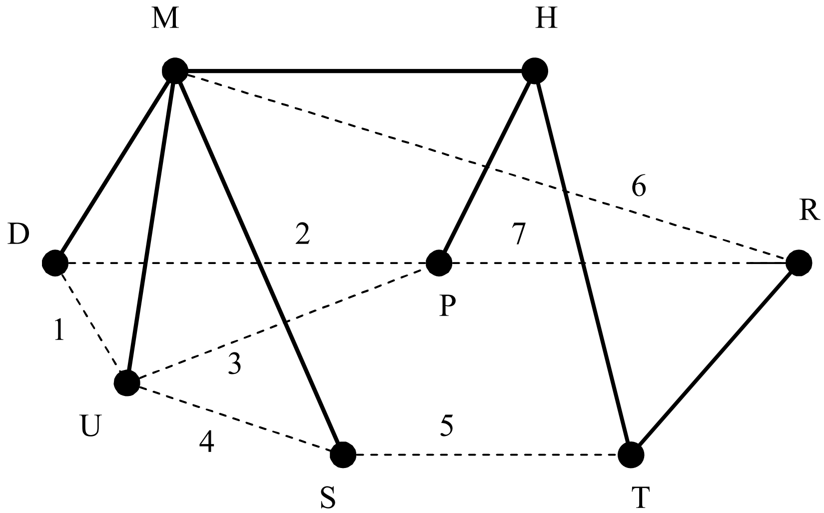
Хроматическое число графа 3.

Определение хроматического класса графа. Раскраска ребер:



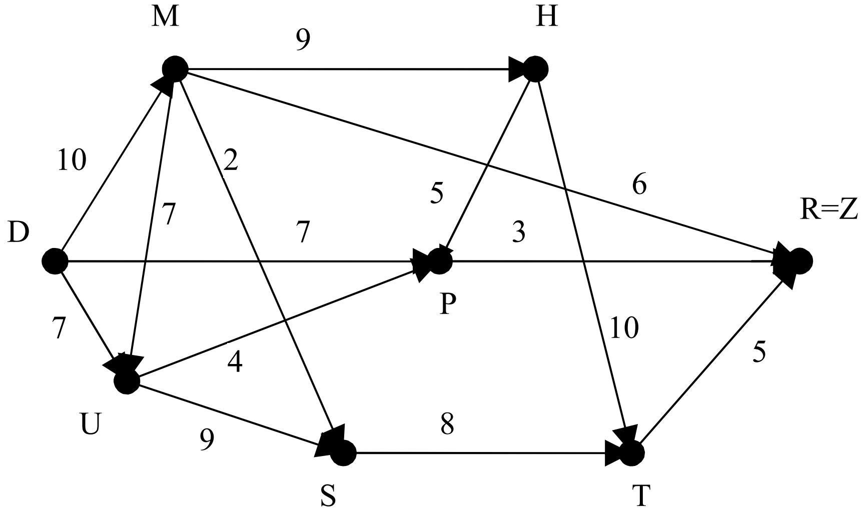
Хроматический класс графа 5.

Определение **цикломатического числа**: 1)14-8+1=7; 2) остов графа:



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

	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

матрица пути

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

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

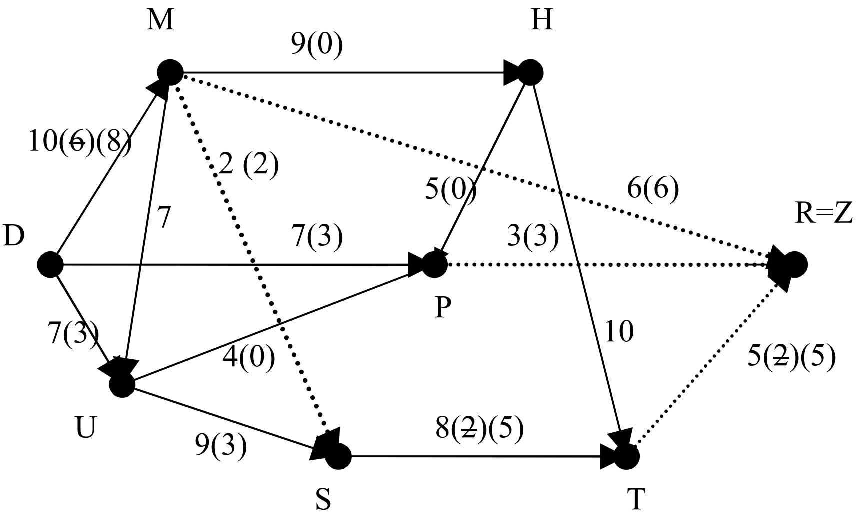
	D	M	U	H	P	S	T	R
D	0	M	U	P	P	M	M	P
M	D	0	U	H	U	S	S	R
U	D	M	0	P	P	S	S	P
H	P	M	P	0	P	M	T	P
P	D	U	U	H	0	U	H	R
S	M	M	U	M	U	M	T	M
T	S	S	S	H	H	S	0	R
R	P	M	P	P	P	M	T	0

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	M	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:

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

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

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

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

Максимальный поток сети: Ф=6+3+5=8+3+3=14.

Эйлерова цепь и цикл.

В данном графе 6 вершин с нечетными степенями, поэтому построение эйлеревой цепи и цикла невозможно.

Задача коммивояжера. Гаммильтонова цепь и цикл.

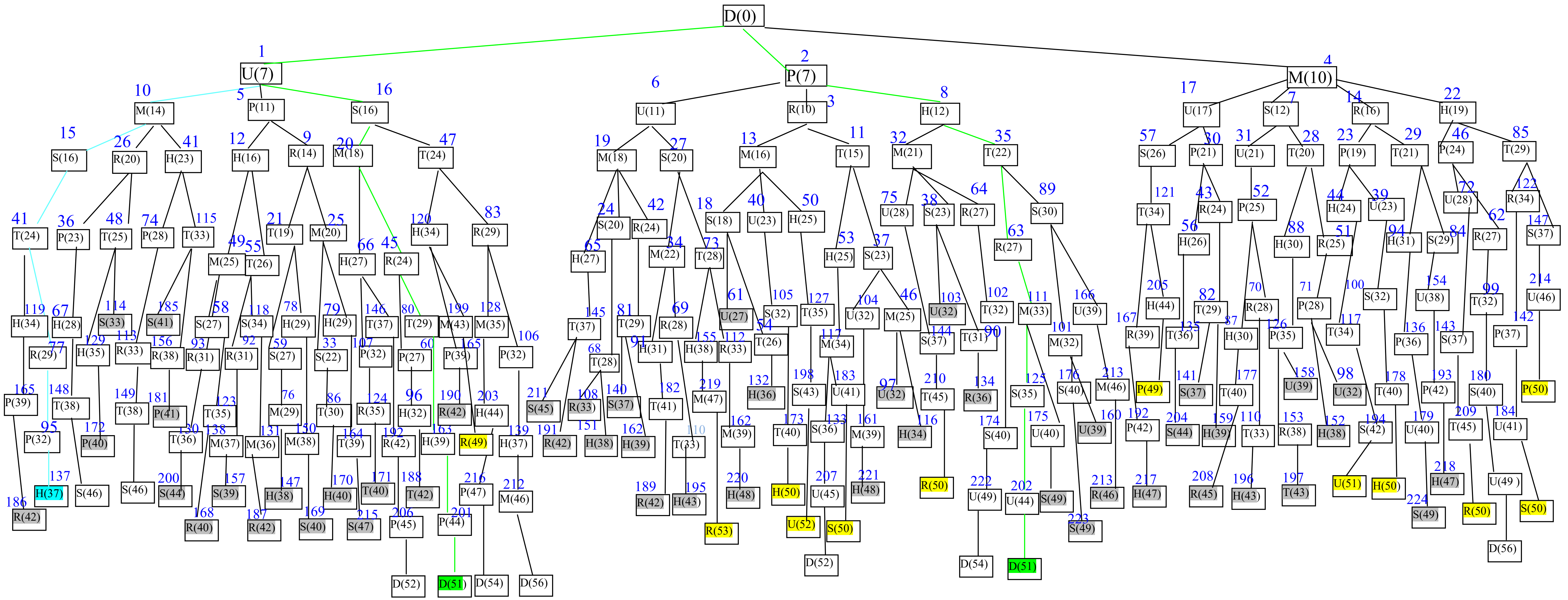
Синим цветом показан порядок выбора вершины с минимальной оценкой.

Желтый фон вершины – отсеченные вершины.

Серый фон вершины – тупиковые вершины.

Зеленый цветом показан самый короткий гаммильтонов цикл (ответ задачи коммивояжера) - 51.

Бирюзовым цветом показан самый гаммильтонов путь – 37 .


D(0)
U(7)
P(7)
M(10)
M(14)
P(11)
S(16)
U(11)
R(10)
H(12)
U(17)
S(12)
R(16)
H(19)

ИСПОЛЬЗУЕМАЯ ЛИТЕРАТУРА

1. Новиков, Ф. А. Дискретная математика для программистов : учеб. для ВУЗов / Ф. А. Новиков. - 3-е изд. – Санкт-Петербург : Питер, 2009. – 384 с.

2. Кузнецов, О. П. Дискретная математика для инженера / О. П. Кузнецов. - 6-е изд., стер. - Санкт-Петербург : Лань, 2009. - 400 с.

3. Яблонский, С. В. Введение в дискретную информатику : учеб. пособие для вузов / С. В. Яблонский. - 4-е изд., стер. – Москва : Высш. шк., 2003. – 384 с.

4. Оре, О. Теория графов / О. Оре ; пер. с англ. И. Н. Врублевской ; под ред. Н. Н. Воробьева. - 2-е изд., стер. – Москва : Наука, 1980. - 336 с.

5. Конспект лекций по курсу "Дискретная математика" [Электронный ресурс] : направление подготовки: 0915 Компьютерная инженерия : специальности: 091501 Компьютерные системы и сети, 091502 Системное программирование / Государственное высшее учебное заведение "Донецкий национальный технический университет", Кафедра компьютерной инженерии ; ГВУЗ "ДонНТУ", Каф. комп. инженерии ; сост. О. Ю. Чередникова. - Электрон. дан. (1 файл : 4 Мб). - Донецк : ГВУЗ "ДонНТУ", 2013. - Систем. требования: ZIP-архиватор.

6. Форд, Л. Р. Потоки в сетях / Л. Р. Форд, Д. Р. Фалкерсон. – Москва : Мир, 1966. – 276 с.

7. Цой, С. Прикладная теория графов / С. Цой, С. М. Цхай. - Алма-Ата : Наука, 1971. – 500 с.

МЕТОДИЧЕСКИЕ УКАЗАНИЯ К РАСЧЕТНОЙ РАБОТЕ ПО ДИСЦИПЛИНЕ «ДИСКРЕТНАЯ МАТЕМАТИКА» (для студентов очной, заочной формы обучения) направления подготовки 09.03.01 «Информатика и вычислительная техника» профилей «Вычислительные машины, комплексы, системы и сети», «Программное обеспечение средств вычислительной техники»

Составитель: Ольга Юрьевна Чередникова, к.т.н., доцент