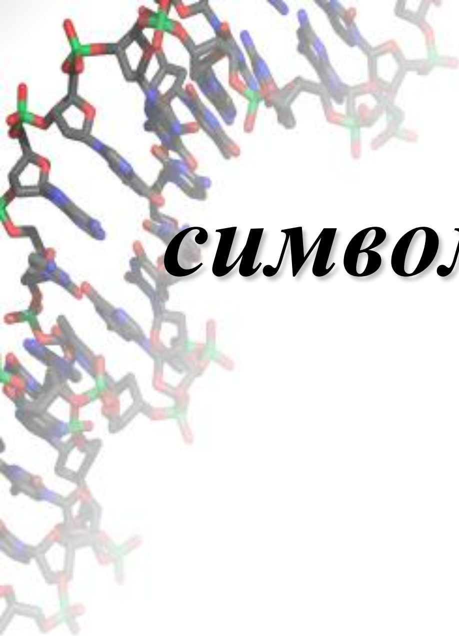


Анализ символьных последовательностей

по мотивам биоинформатики

М.А. Ройтберг

Занятие 3. Часть 1

ГИПЕРГРАФЫ

(окончание)

Яндекс

2 марта 2015

ПОВТОРЕНИЕ
ОРИЕНТИРОВАННЫЕ ГИПЕРГРАФЫ
(ОГ-ГРАФЫ)

1. Гиперграф, гиперребро

Ориентированный гиперграф (ОГ-граф) над полукольцом A – это
четверка $G = (V, q_0, E, f)$, где
- V – конечное множество вершин;
- $q_0 \in V$ – *стартовая* вершина;
- E – конечное множество *гиперребер,*
  т.е. пар вида (v, W), где
  - $v \in V$,
  - W – упорядоченный список вершин из V;
- $f: E \to A$ – функция пометок на ребрах.

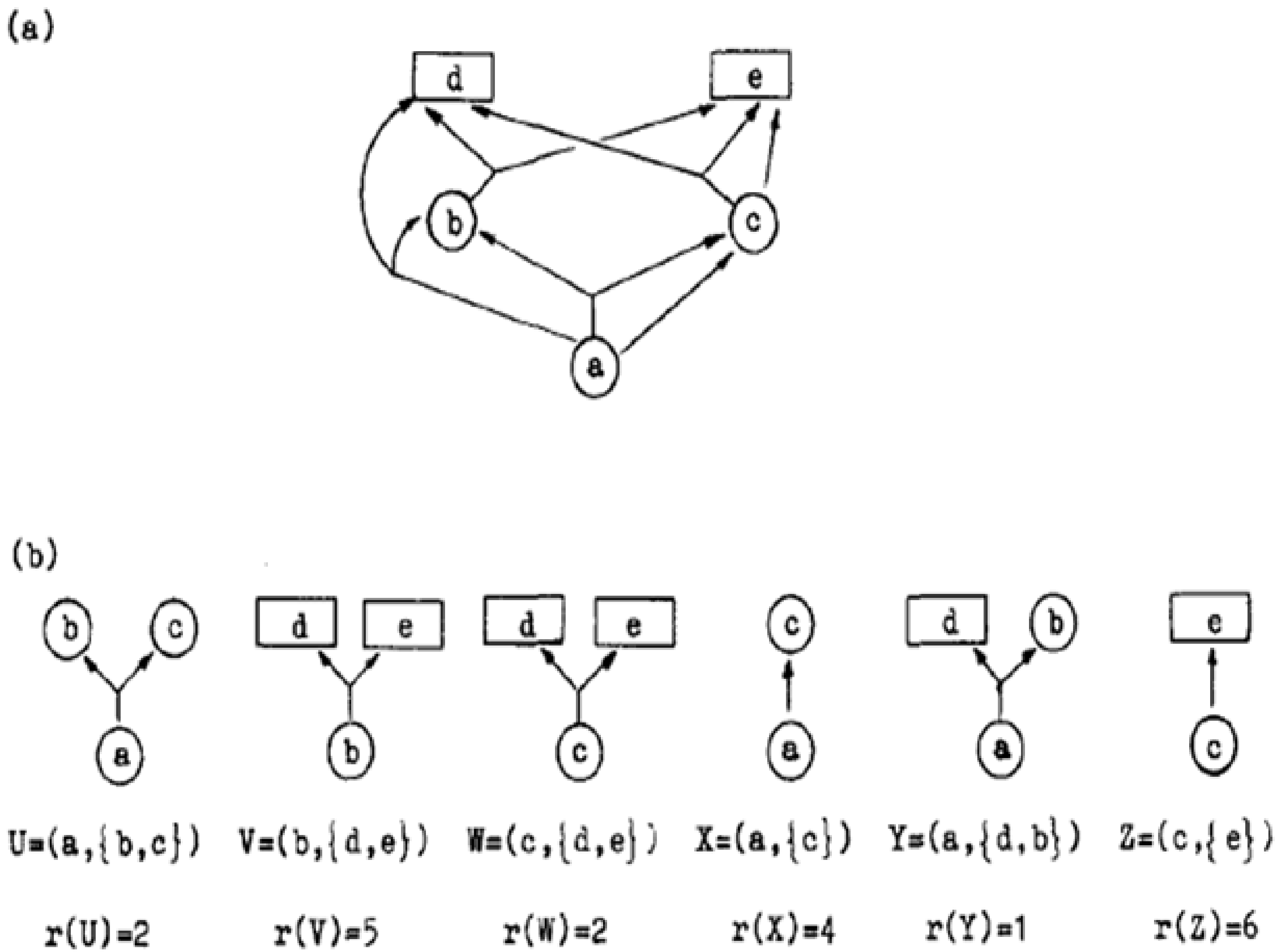
2. Гиперребро. Начальная вершина. Конечные вершины.

Вершина *v* называется *начальной* вершиной гиперребра *e=(v, W)*, вершины из *W* называются конечными вершинами этого гиперребра; значение *f(e)* $\in$ *A* называется *весом* гиперребра *e.*

Вершина v ОГ-графа G называется терминальной, если в G нет гиперребра, в котором v является начальной вершиной.

Аналог пути в графе - гиперпуть в гиперграфе

Пример гиперграфа
5 вершин; 6 гиперребер



Аналог пути в графе - гиперпуть в гиперграфе

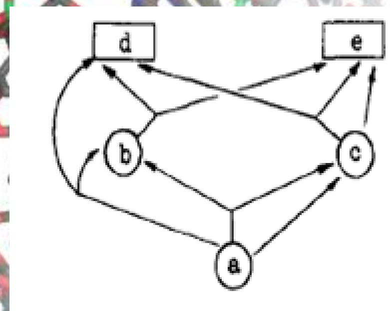
3. Гиперпуть.

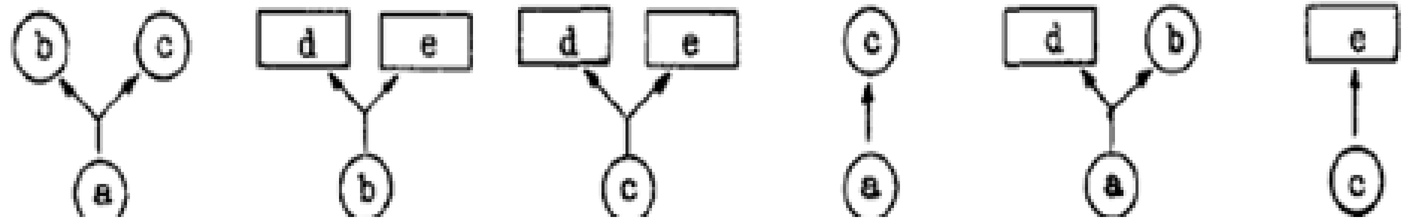
Гиперпуть в ОГ-графе G = (V, q0, E, f) –это такое конечное упорядоченное помеченное дерево T, что

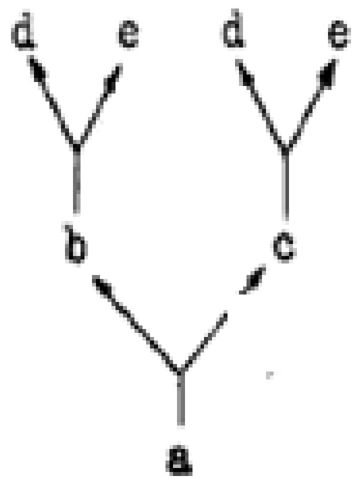
1) каждый узел дерева T помечен вершиной ОГ-графа G; одна вершина v из V может соответствовать нескольким узлам дерева T.

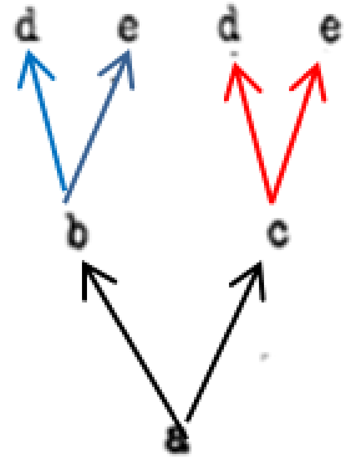
2) каждому внутреннему узлу r дерева T соответствует гиперребро e = (v, W), причем узел r помечен вершиной v, а список пометок в сыновьях узла r, взятых в порядке, предписанном деревом T, совпадает со списком W.

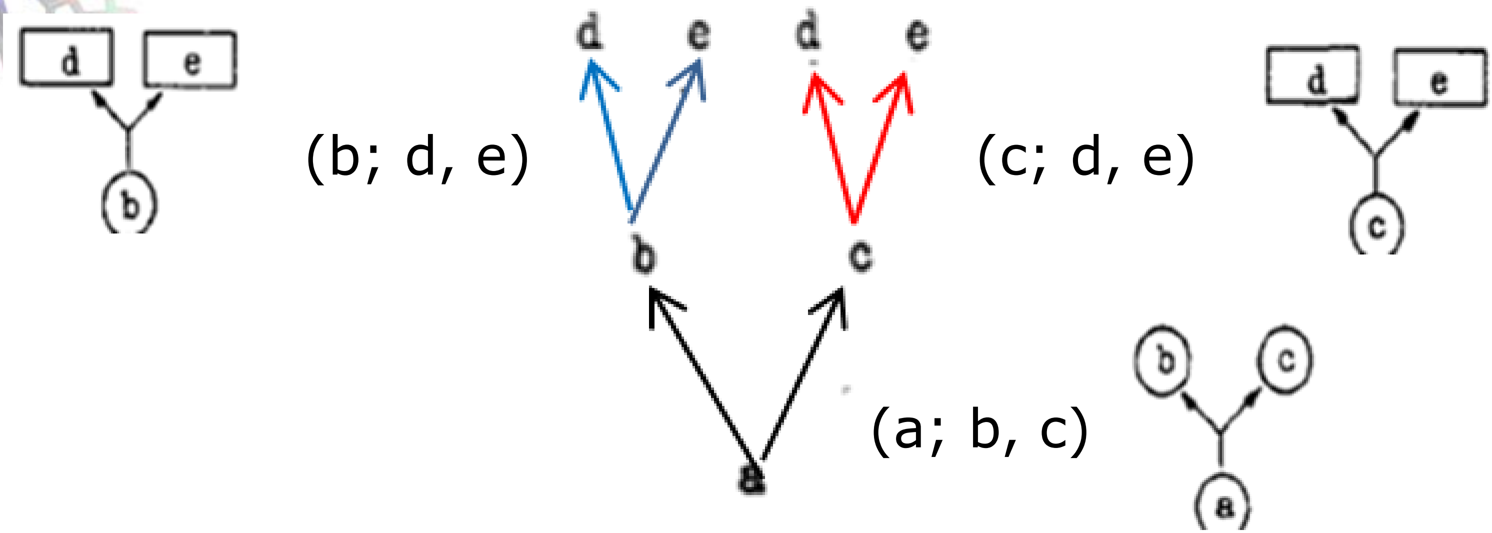
Корневое гиперребро гиперпути *T* – гиперребро, соответствующее корню *T.*
Начальная (корневая) вершина – вершина, соответствующее корню *T.*


d
e
b
c
a


b c
a
d e
b
d e
c
c
a
d b
a
e
c


d e d e
b c
a

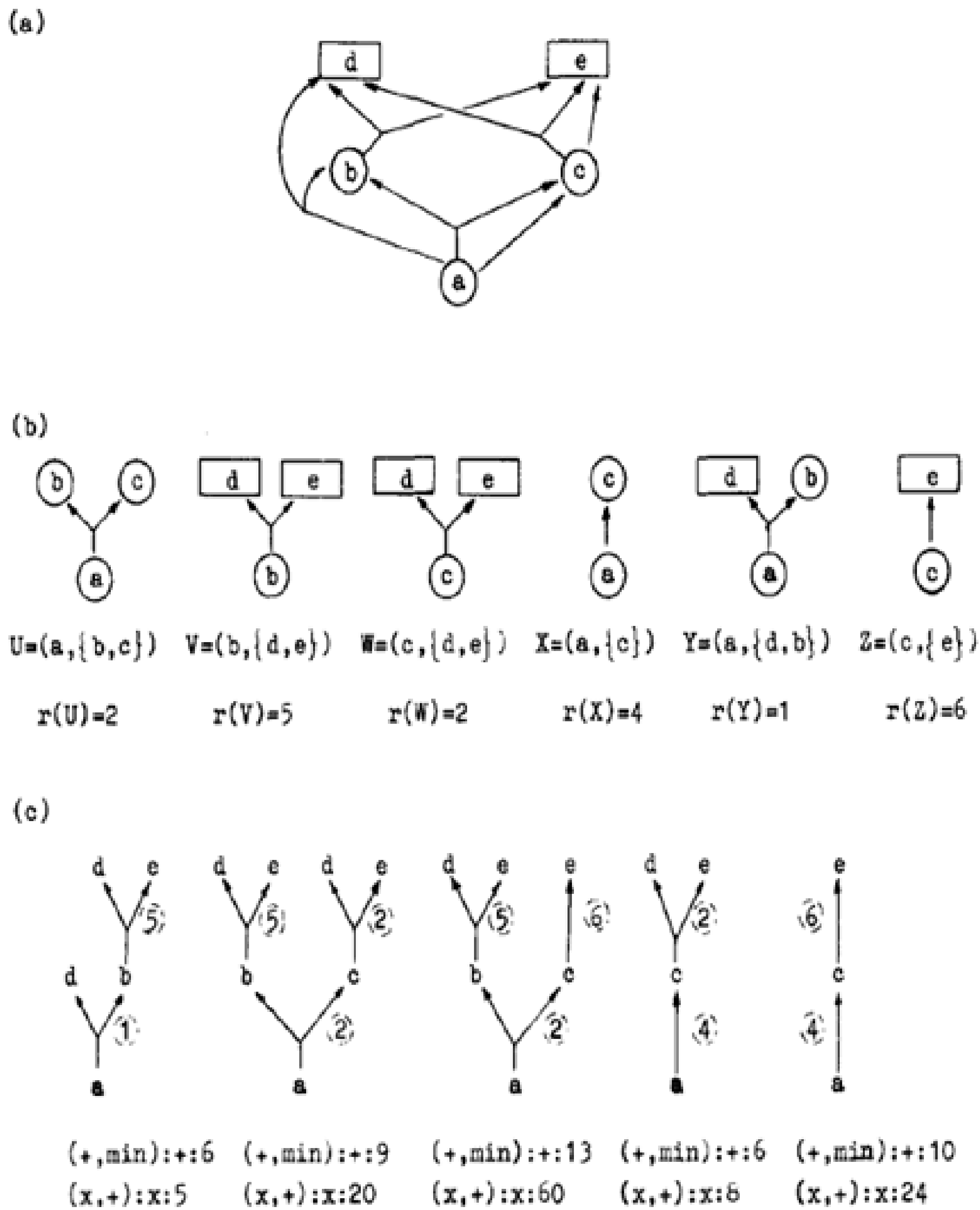

d e d e
b c
a



Гиперпуть в ОГ-графе G = (V, q0, E, f) –это такое конечное упорядоченное помеченное дерево T, что

1) каждый узел дерева T помечен вершиной ОГ-графа G; одна вершина v из V может соответствовать нескольким узлам дерева T.

2) каждому внутреннему узлу (= не листу) r дерева T соответствует гиперребро e = (v, W), причем узел r помечен вершиной v, а список пометок в сыновьях узла r, взятых в порядке, предписанном деревом T, совпадает со списком W.



(a)
d
e
b
c
a
(b)
U=(a,{b,c})
V=(b,{d,e})
W=(c,{d,e})
X=(a,{c})
Y=(a,{d,b})
Z=(c,{e})
r(U)=2
r(V)=5
r(W)=2
r(X)=4
r(Y)=1
r(Z)=6
(c)
(+,min):+:6
(x,+):x:5
(+,min):+:9
(x,+):x:20
(+,min):+:13
(x,+):x:60
(+,min):+:6
(x,+):x:6
(+,min):+:10
(x,+):x:24

4. Ацикличность

- Определение 1. ОГ-граф $G = (V, q_0, E)$ называется *ациклическим,* если в G не существует гиперпути, в котором корень помечен той же вершиной, что и еще какой-либо узел этого гиперпути.
- Утверждение 1. В *ациклическом* ОГ-графе существует лишь конечное число различных потоков.
- Определение 2. ОГ-граф $G = (V, q_0, E)$ называется *сильно ациклическим,* если в G не существует гиперпути, в котором два разных узла помечены одной и той же вершиной.
- Утверждение 2. Всякий сильно ациклический ОГ-граф является ациклическим.

4. Ацикличность

- Определение 1. ОГ-граф $G = (V, q_0, E)$ называется *ациклическим,* если в G не существует гиперпути, в котором корень помечен той же вершиной, что и еще какой-либо узел этого гиперпути.
- Утверждение 1. В *ациклическом* ОГ-графе существует лишь конечное число различных потоков.
- Определение 2. ОГ-граф $G = (V, q_0, E)$ называется *сильно ациклическим,* если в G не существует гиперпути, в котором два разных узла помечены одной и той же вершиной.
- Утверждение 2. Всякий сильно ациклический ОГ-граф является ациклическим.

Вес R*(T)* *гиперпути T* (аналог веса пути на графах) определяется рекурсивно.

5. Вес гиперпути: рекурсивное определение

- Пусть $G = (V, q_0, E, f)$ – ОГ-граф; T – гиперпуть в G. Определим *вес $R(T)$ гиперпути T* следующим образом.
- Если гиперпуть T состоит из единственного узла, то $R(T) = i$, где i – нейтральный элемент относительно операции * (умножения).
- Пусть корень T помечен гиперребром $e = (v, W)$; $x_1, \ldots, x_N$ – упорядоченный список сыновей корня дерева T; T_k – поддерево дерева T с корнем в узле x_k ($k = 1, \ldots, N = |W|$). Тогда

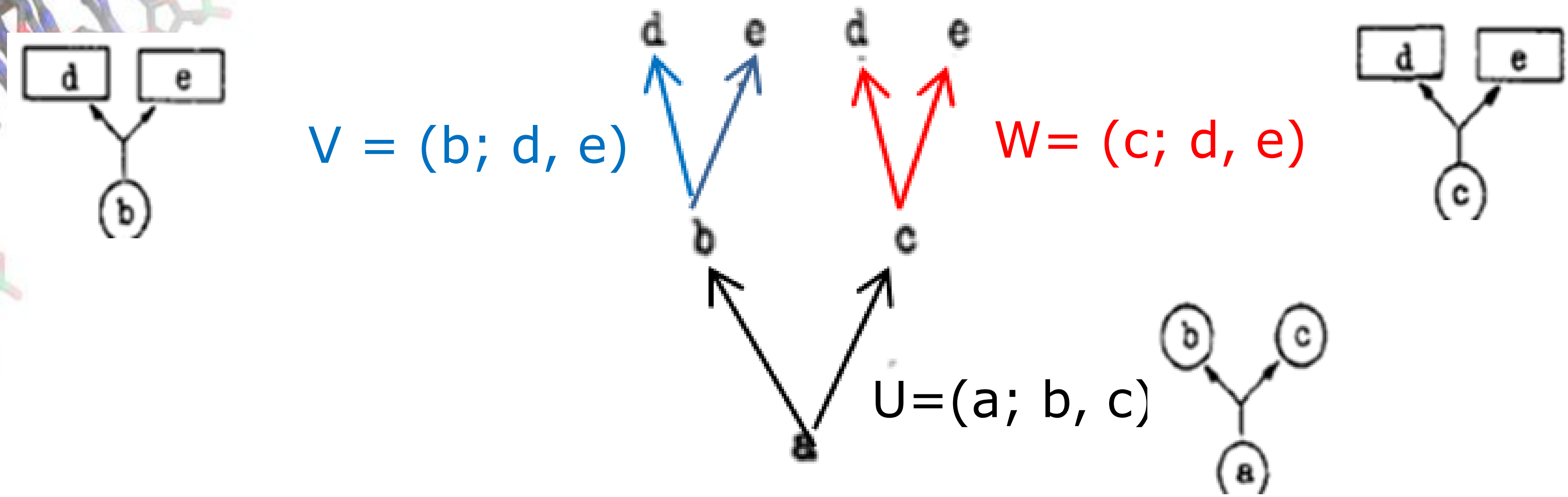
$$R(T) = f(e) * R(T_1) * \ldots * R(T_N)$$

5. Вес гиперпути: явное определение

- Пусть $G = (V, q_0, E, f)$ – ОГ-граф; T – гиперпуть в G. Пусть корень T помечен гиперребром $e = (v, W)$; $x_1, \ldots, x_N$ – упорядоченный список сыновей корня дерева T; T_k – поддерево дерева T с корнем в узле x_k ($k =1, \ldots, N = |W|$).Тогда

$$R(T) = f(e) * R(T_1) * \ldots * R(T_N)$$

Вес гиперпути T это произведение весов гиперребер (в смысле операции *), соответствующих узлам T, причем порядок перемножения соответствует *левому обходу* дерева T в глубину.



$$R(T) = f(U)*f(V)*f(W)$$

6. Терминальные и полные пути

- Гиперпуть *T* в ОГ-графе *G* называется *терминальным,* если все его листья соответствуют терминальным вершинам ОГ-графа *G*.

- Терминальный гиперпуть называется *полным,* если его корню соответствует стартовая вершина ОГ-графа *G*.

7. Основная задача на гиперграфах

- Гиперпуть *T* в ОГ-графе *G* называется *терминальным,* если все его листья соответствуют терминальным вершинам ОГ-графа *G*.
- Терминальный гиперпуть называется *полным,* если его корню соответствует стартовая вершина ОГ-графа *G*.

- Определение 6.2. *Обобщенной статистической суммой* ОГ-графа *G* называется сумма (в смысле операции +) *S(G)* весов всех его полных гиперпутей.

- ***Задача.* Найти обобщенную статистическую сумму для заданного ациклического ОГ-графа G**

8. Решение основной задача на гиперграфах *Найти обобщенную статистическую сумму для заданного ациклического ОГ-графа G*

- *Обозначения. Пусть $v \in V$.*
- G_v – это подграф ОГ-графа *G,* порожденный всеми вершинами *G,* достижимыми из *v,* т.е. вершинами, которые встречаются в гиперпутях, корень которых соответствует вершине *v.*
- *S(V) – значение обобщенной статистической суммы для гиперграфа G_v – т.е. сумма весов всех полных путей в G_v*

Общее описание алгоритма

- Перебираем вершины *ОГ-графа G* в обратном топологическом порядке и для каждой вершины $v \in V$ вычисляем S(V*)*.
- Для любой терминальной вершины *v* положим *S(v)=i,* где *i* – правый нейтральный элемент относительно «умножения».
-

Вычисление S(v*)* для внутренней вершины v.

- $e_1 = <v; x_1, \ldots, x_k>$, $e_2 = <v; y_1, \ldots, y_m>$,
- $e_3 = <v; z_1, \ldots, z_n>$ - все гиперребра с начальной вершиной *v*
- $\boldsymbol{H_v}$– множество всех гиперпутей с начальной вершиной *v*
- $\boldsymbol{H_v(e)}$ – множество всех гиперпутей с корневым гиперребром ***e*** и начальной вершиной ***v***
  - $\boldsymbol{H(v) = H_v(e_1) + H_v(e_2) + H_v(e_3)}$
- *Пусть* $\boldsymbol{S_v(e_r)}$ *обозначает сумму весов всех гиперпутей из* $H_v(e_r)$, $r = 1, 2, 3$.
  - $$\boldsymbol{S(v) = S_v(e_1) + S_v(e_2) + S_v(e_3)}$$
-

Вычисление S(v) для внутренней вершины v. Окончание

- $e_1 = <v; x_1, \ldots, x_k>$, $e_2 = <v; y_1, \ldots, y_m>$,
- $e_3 = <v; z_1, \ldots, z_n>$ - все гиперребра с начальной вершиной *v*

  - $S(v) = S_v(e_1) + S_v(e_2) + S_v(e_3)$
  - $S_v(e_1) = f(e_1)* S(x_1)*\ldots* S(x_k)$

    $S_v(e_2) =\ldots;\quad S_v(e_3) = \ldots$

$$S(v) = f(e_1)* S(x_1)*\ldots* S(x_k) + $$
$$+ f(e_2)* S(y_1)*\ldots* S(y_m) +$$
$$+ f(e_3)* S(z_1)*\ldots* S(z_n)$$

-

Время работы алгоритма

Предполагаем: время перехода к обработке очередной вершины и время доступа к ранее вычисленным суммам не зависят от размера графа.

$$\boldsymbol{S(v) = f(e_1) * S(x_1) * \ldots * S(x_k) + f(e_2) * S(y_1) * \ldots * S(y_m) + f(e_3) * S(z_1) * \ldots * S(z_n)}$$

T – сумма времен обработки гиперребер. Время обработки гиперрребра ~ количества его концевых вершин => ***T*** ~ суммарное количество концевых вершин всех гиперребер.

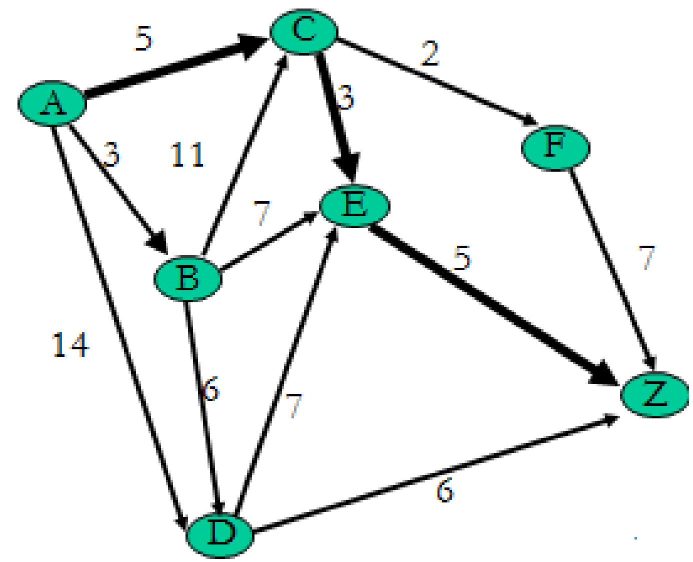
- Если степени всех гиперребер ограничены, то

T ~ количество гиперребер.

9. Что делать, если обратный топологический порядок на множестве вершин неизвестен?

9. Что делать, если ….
А. Графы (минимальный путь)

- Для каждой вершины знаем:
  - все входящие ребра;
  - количество исходящих ребер.



Вершина готова к обработке, если обработаны все ее исходящие ребра.

Начало: готовы к обработке терминальные вершины

Вершины	A	B	C	D	E	F	Z
К-во необр. исход. ребер	3	3	2	2	1	1	0
Тек. вес мин. пути	∞	∞	∞	∞	∞	∞	0

Список вершин, ждущих обработки	Z

9. Что делать, если обратный топологический порядок неизвестен?
А. Графы (минимальный путь)

Обработка вершины *v.*

ДЛЯ КАЖДОЙ вершины u ТАКОЙ, ЧТО

∃ ребро (u; v){

{ уменьшить кол-во необр. ребер для u;

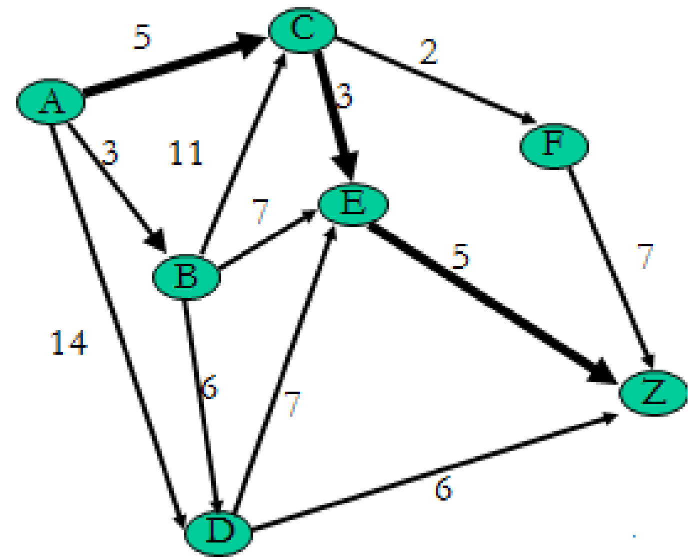
перевычислить вес мин. пути для u;

ЕСЛИ больше нет необр. ребер

ТО добавить u в список вершин

готовых к обработке

}

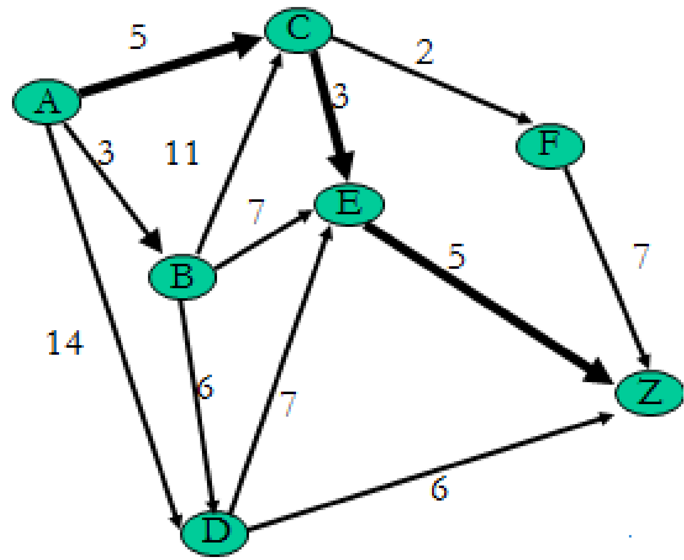


Вершины	A	B	C	D	E	F	Z
К-во необр. исход. ребер	3	3	2	1	0	0	0
Тек. вес мин. пути	∞	∞	∞	6	5	7	0

Список вершин, ждущих обработки	E, F

9. Что делать, если обратный топологический порядок неизвестен?
А. Графы (минимальный путь)

Обработка вершины *v.*
ДЛЯ КАЖДОЙ вершины u ТАКОЙ, ЧТО
∃ ребро (u; v){
{ уменьшить кол-во необр. ребер для u;
перевычислить вес мин. пути для u;
ЕСЛИ больше нет необр. ребер
ТО добавить u в список вершин
готовых к обработке
}

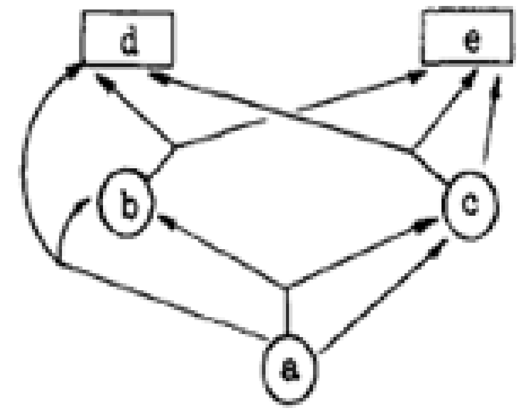


Вершины	A	B	C	D	E	F	Z
К-во необр. исход. ребер	3	2	1	0	0	0	0
Тек. вес мин. пути	∞	12	8	6	5	7	0

Список вершин, ждущих обработки	F, D

9. Что делать, если ….
Б. Гиперграфы (минимальный путь)

- Для каждой ***вершины*** знаем:
  - все входящие гиперребра (=г.р., у которых v – конечная);
  - количество исходящих ребер.
- Для каждого ***гиперребра*** знаем:
  - количество конечных вершин (без учета кратностей)



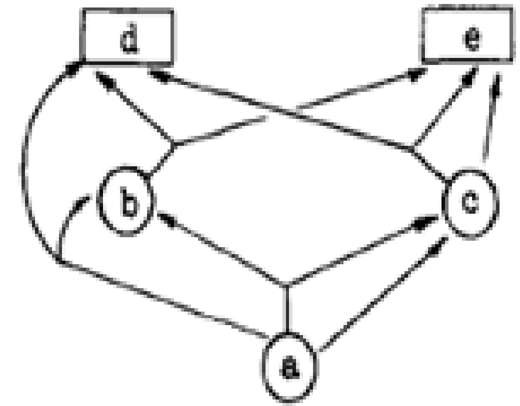
Вершины	A	B	C	D	E
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин	2	2	1	2	2	1

Список вершин, ждущих обработки	D, E

9. Что делать, если ….
Б. Гиперграфы (минимальный путь)

- Для каждой вершины знаем:
  - все входящие ребра;
  - все входящие гиперребра (=г.р., у которых v – конечная);
  - количество исходящих ребер.



Вершина готова к обработке, если обработаны все ее исходящие гиперребра.

Начало: готовы к обработке терминальные вершины

Вершины	A	B	C	D	E
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин	2	2	1	2	2	1

Список вершин, ждущих обработки	D, E

9. Что делать, если обратный топологический порядок неизвестен?
А. Графы (минимальный путь)

Обработка вершины *v*.

ДЛЯ КАЖДОЙ вершины u ТАКОЙ, ЧТО

∃ ребро (u; v){

{ уменьшить кол-во необр. ребер для u;

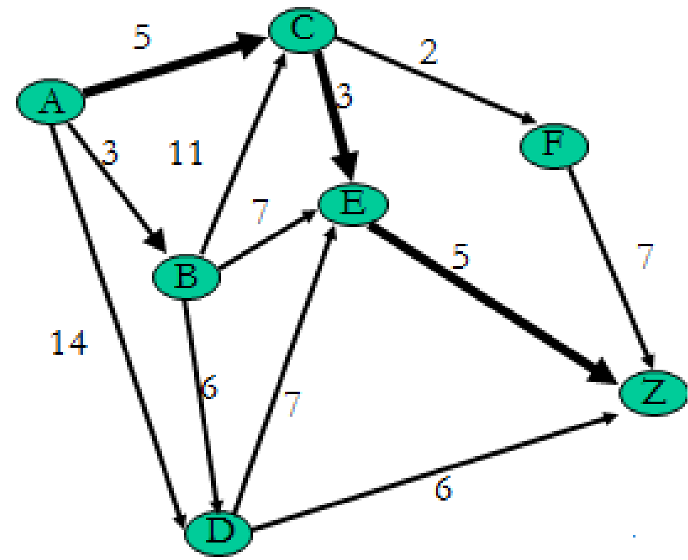
перевычислить вес мин. пути для u;

ЕСЛИ больше нет необр. ребер

ТО добавить u в список вершин

готовых к обработке

}



Вершины	A	B	C	D	E	F	Z
К-во необр. исход. ребер	3	3	2	1	0	0	0
Тек. вес мин. пути	∞	∞	∞	6	5	7	0

Список вершин, ждущих обработки	E, F

9. Что делать, если обратный топологический порядок неизвестен?
А. Графы (минимальный путь)

Обработка вершины *v.*

ДЛЯ КАЖДОГО ребра (u; v){

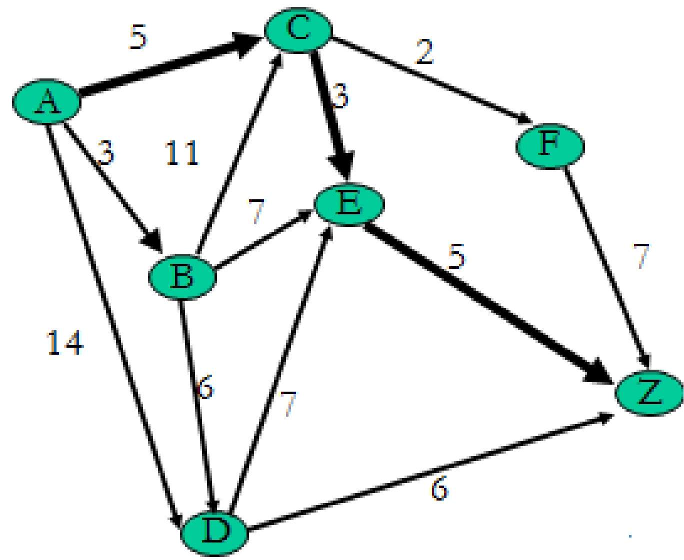
{ уменьшить кол-во необр. ребер для u;

перевычислить вес мин. пути для u;

ЕСЛИ больше нет необр. ребер

ТО добавить u в список вершин готовых к обработке

}



Вершины	A	B	C	D	E	F	Z
К-во необр. исход. ребер	3	3	2	1	0	0	0
Тек. вес мин. пути	∞	∞	∞	6	5	7	0

Список вершин, ждущих обработки	E, F

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)

Обработка вершины *v*.

ДЛЯ КАЖДОГО гиперребра h=(u; {…,v, …})

{уменьшить кол-во W(h);

ЕСЛИ W(h) = 0 ТО

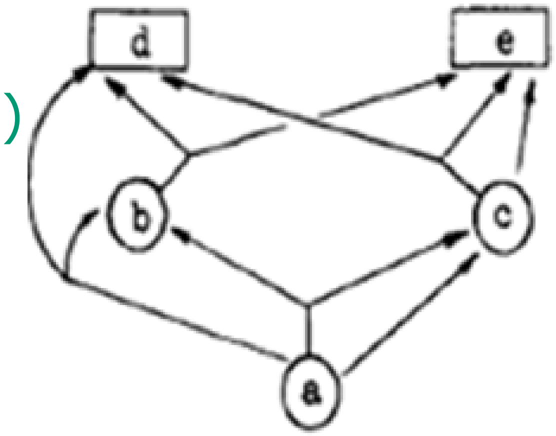
уменьшить кол-во необр. ребер для u;

перевычислить вес мин. пути для u;

ЕСЛИ для u больше нет необр. ребер

ТО добавить u в список вершин готовых к обработке

}



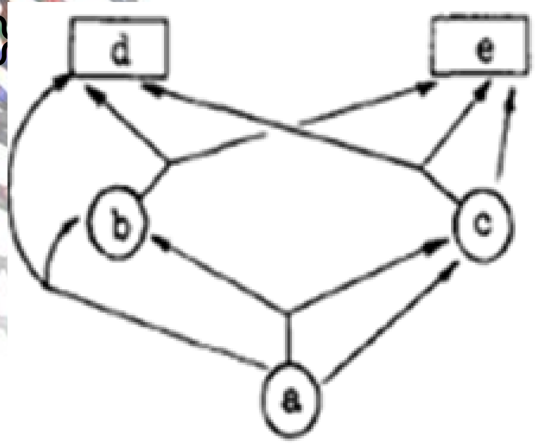
Вершины	A	B	C	D	E
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

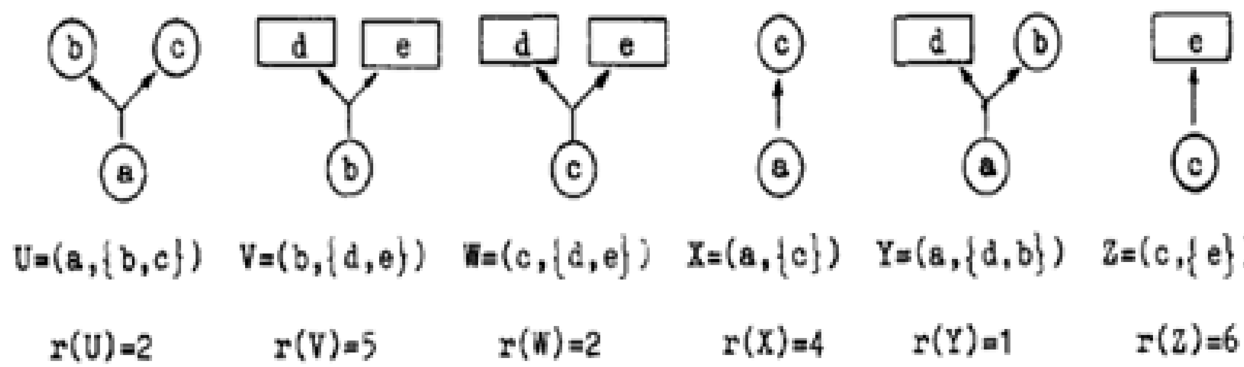
Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин	2	2	1	2	2	1

Список вершин, ждущих обработки	D, E

9. Что делать, если обратный топологический порядок неизвестен?

Б. Гиперграфы (минимальный путь)



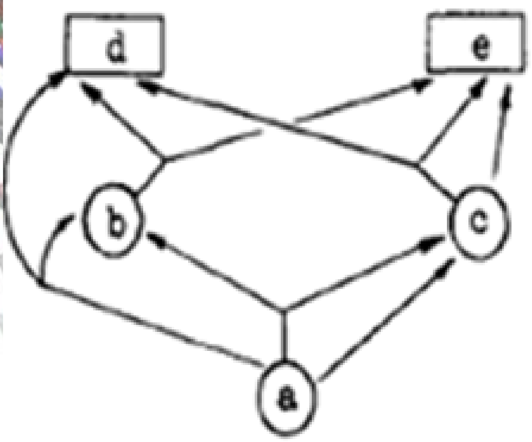


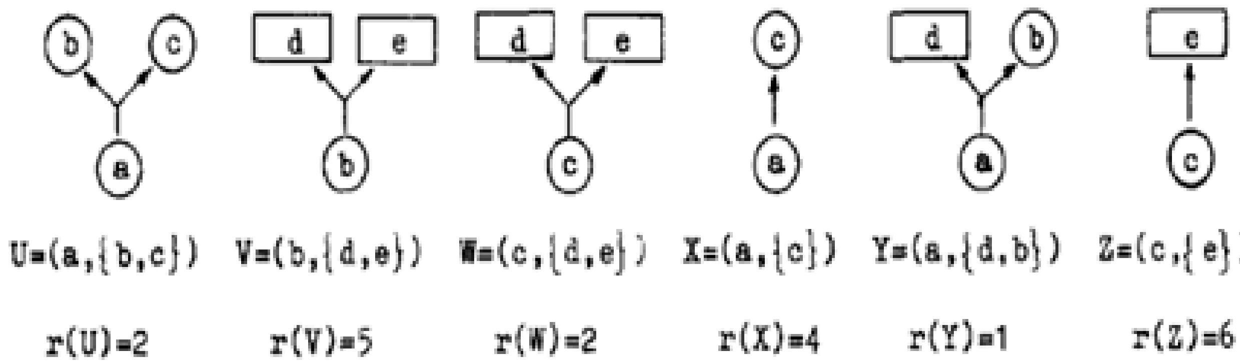
Вершины	**A**	**B**	**C**	**D**	**E**
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

Гиперребра	**A; B, C**	**A; D, B**	**A; C**	**B; D, E**	**C; D, E**	**C; E**
К-во необр. конечных вершин	2	2	1	2	2	1

Список вершин, ждущих обработки	D, E

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)





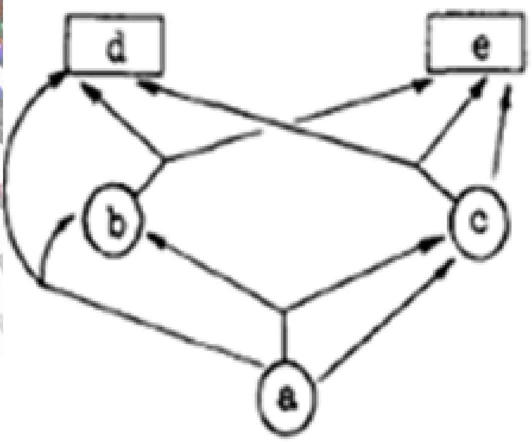
Обрабатываем вершину D

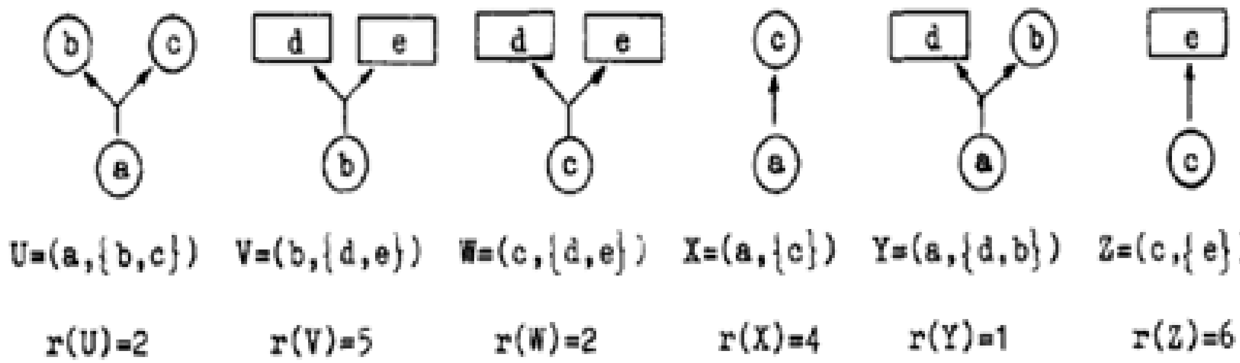
Вершины	**A**	**B**	**C**	**D**	**E**
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

Гиперребра	**A; B, C**	**A; D, B**	**A; C**	**B; D, E**	**C; D, E**	**C; E**
К-во необр. конечных вершин	2	**1**	1	**1**	**1**	1

Список вершин, ждущих обработки	E

9. Что делать, если обратный топологический порядок неизвестен? *Б. Гиперграфы (минимальный путь)*





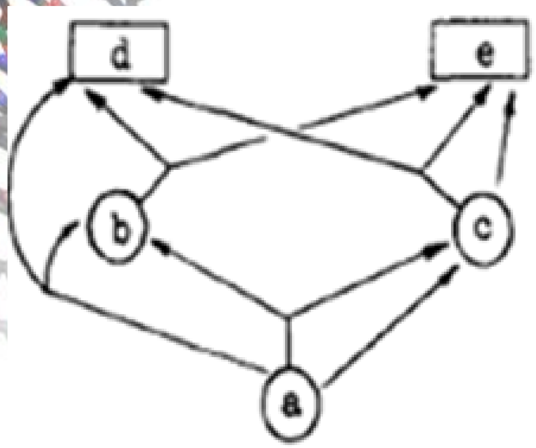
Обрабатываем вершину E: (1) гиперребро (B; (D, E))

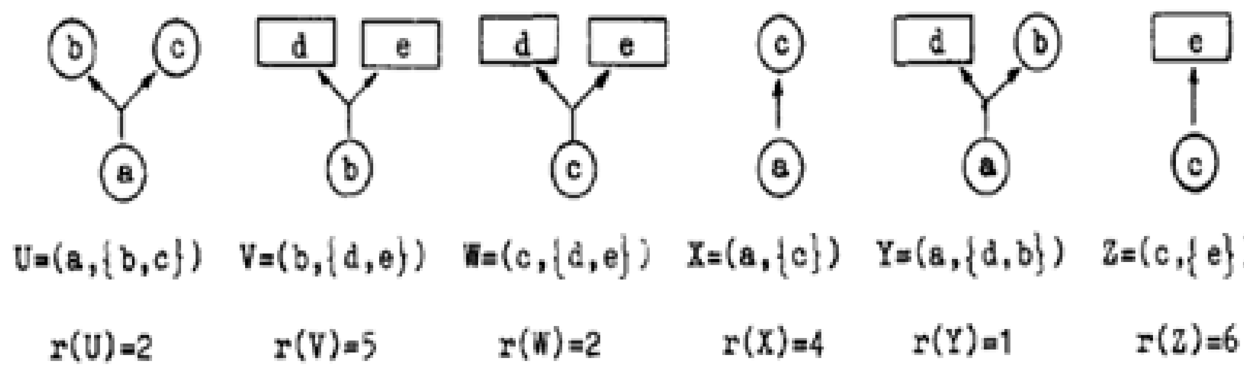
Вершины	A	B	C	D	E
К-во необр. Исход. ГИПЕРребер	3	1	2	0	0
Тек. вес мин. пути	∞	∞	∞	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин	2	1	1	0	1	1

Список вершин, ждущих обработки	E

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)





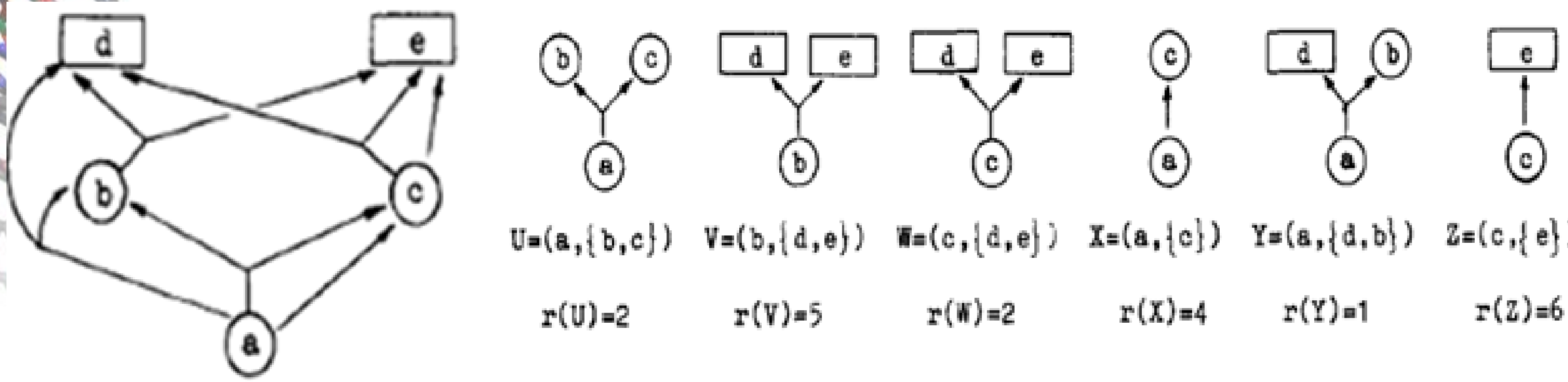
Обрабатываем E: 1) гиперребро (B; (D, E)) => вершина B (1) перевыч. S; (2) H--; (3) H=0 => добавить B в СПИСОК

Вершины		**A**	**B**	**C**	**D**	**E**
К-во необр. исход. ГИПЕРребер (H)		3	0	2	0	0
Тек. вес мин. Пути (S)		∞	5	∞	0	0

Гиперребра		**A; B, C**	**A; D, B**	**A; C**	**B; D, E**	**C; D, E**	**C; E**
К-во необр. конечных вершин (T)		2	1	1	0	1	1

Список вершин, ждущих обработки		E, B

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)



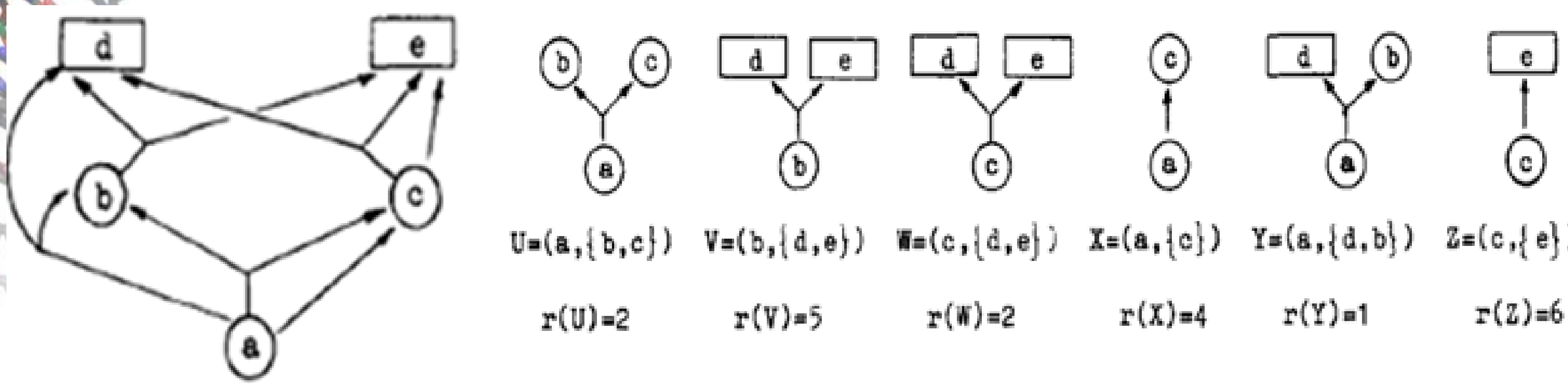
Обрабатываем E: 2) гиперребро (C; (D, E)) => вершина C (1) перевыч. S; (2) H--;

Вершины		A	B	C	D	E
К-во необр. исход. ГИПЕРребер (H)		3	0	1	0	0
Тек. вес мин. Пути (S)		∞	5	2	0	0

Гиперребра		A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (T)		2	1	1	0	0	1

Список вершин, ждущих обработки		E, B

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)



Обрабатываем Е: 3) гиперребро (С; (Е)) => вершина С

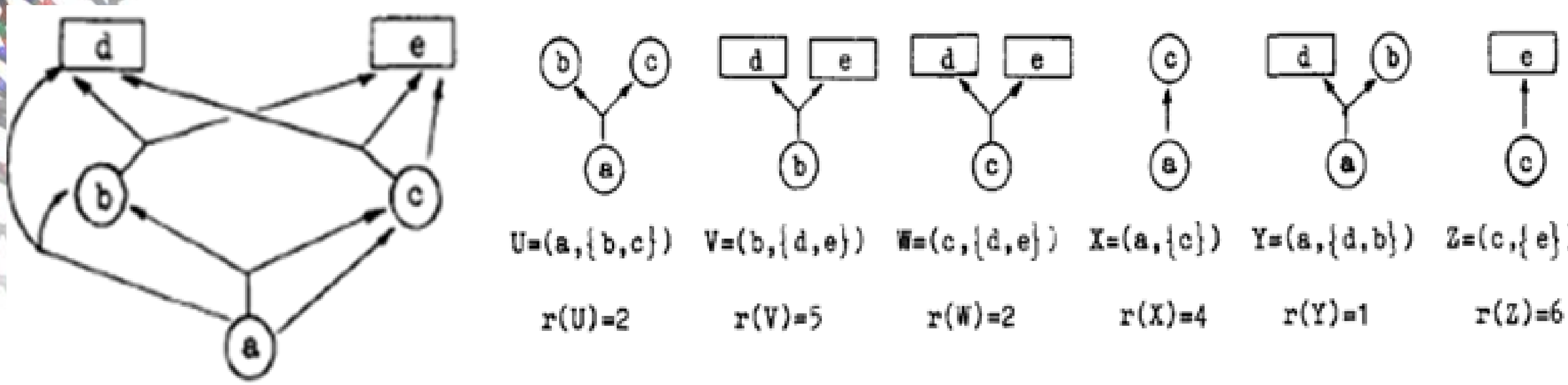
(1) перевыч. S; (2) Н--; (3) Н=0 => добавить С в СПИСОК

Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (Н)	3	0	0	0	0
Тек. вес мин. Пути (S)	∞	5	2	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (Т)	2	1	1	0	0	0

Список вершин, ждущих обработки	E, B, C

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)



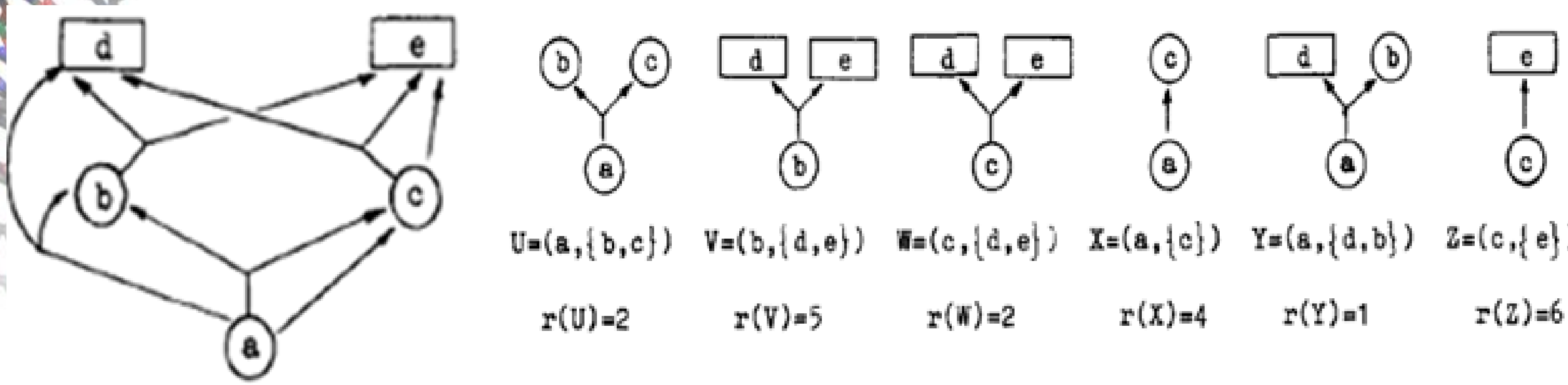
Обрабатываем В: 3) гиперребро (А; (В, С))

Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (Н)	3	0	0	0	0
Тек. вес мин. Пути (S)	∞	5	2	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (T)	1	1	1	0	0	0

Список вершин, ждущих обработки	B, C

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)



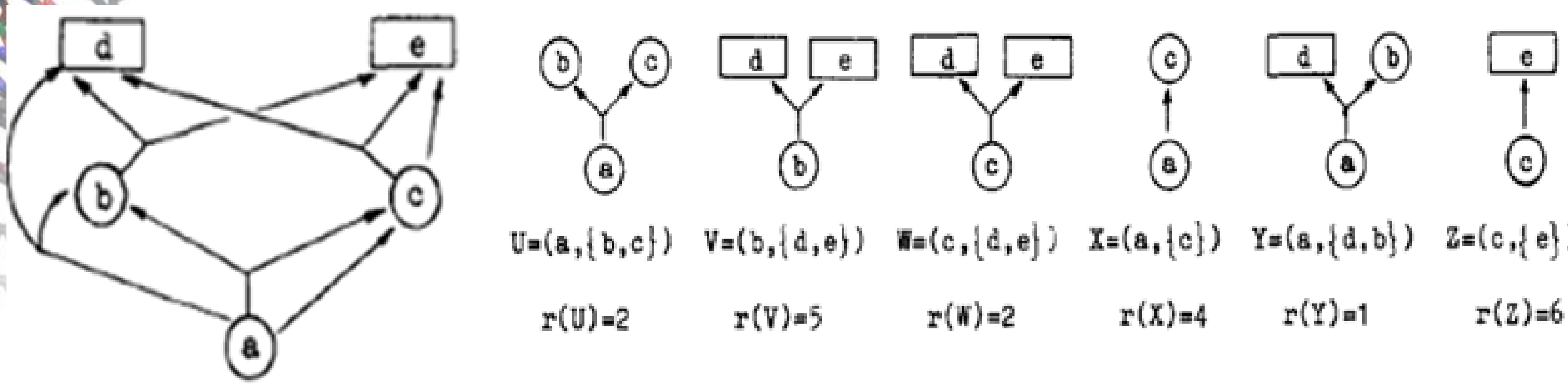
Обрабатываем B: 3) гиперребро (A; (D, B)) => вершина А
(1) перевыч. S; (2) Н--;

Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (H)	**2**	0	0	0	0
Тек. вес мин. Пути (S)	1+5+0=6	5	2	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (T)	1	**0**	1	0	0	0

Список вершин, ждущих обработки	B, C

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)



Обрабатываем С: 3) гиперребро (A; (B, C)) => вершина A

(1) перевыч. S; (2) H--;

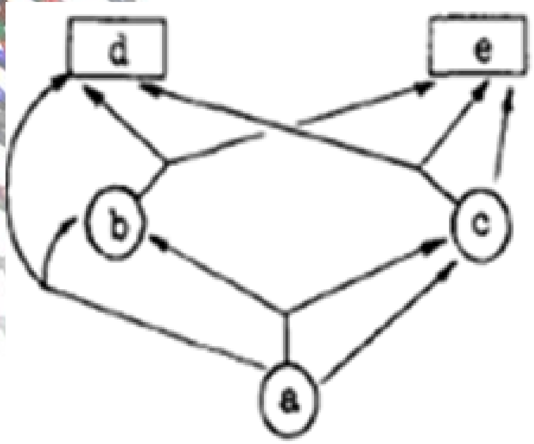
Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (H)	1	0	0	0	0
Тек. вес мин. Пути (S)	2+5+2 > **6**	5	2	0	0

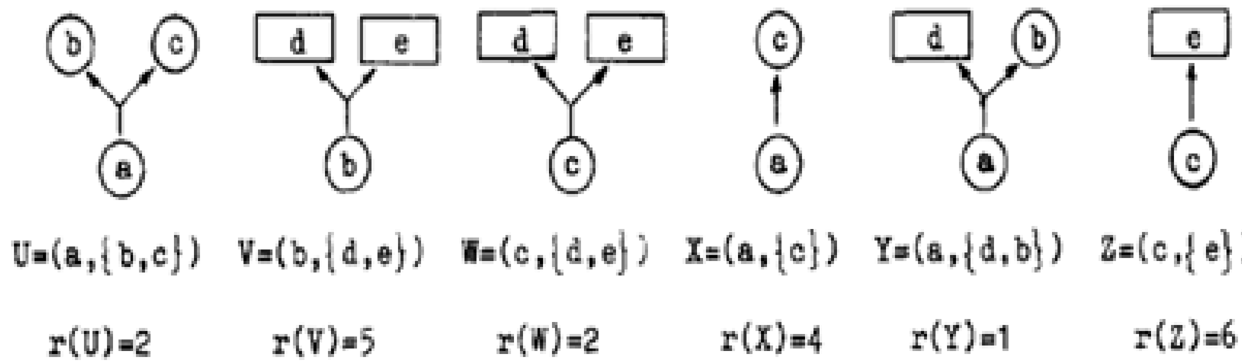
Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (T)	0	0	1	0	0	0

Список вершин, ждущих обработки	C

9. Что делать, если обратный топологический порядок неизвестен?

Б. Гиперграфы (минимальный путь)





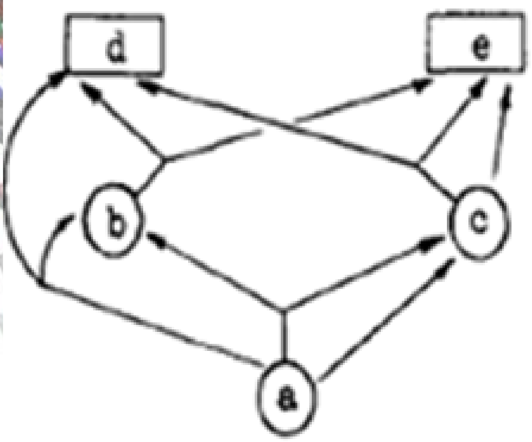
Обрабатываем С: 3) гиперребро (А; (С)) => вершина А
(1) перевыч. S; (2) Н--; (3) Н=0 => добавить А в СПИСОК

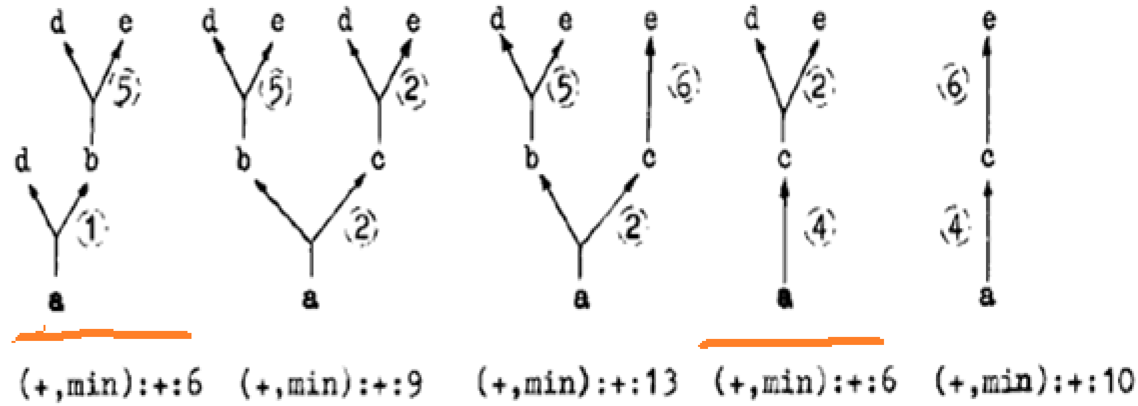
Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (Н)	0	0	0	0	0
Тек. вес мин. Пути (S)	4+2 = **6**	5	2	0	0

Гиперребра	А; В, С	А; D, В	А; С	В; D, Е	С; D, Е	С; Е
К-во необр. конечных вершин (Т)	0	0	0	0	0	0

Список вершин, ждущих обработки	С, А

9. Что делать, если обратный топологический порядок неизвестен?
Б. Гиперграфы (минимальный путь)





Вершины	A	B	C	D	E
К-во необр. исход. ГИПЕРребер (H)	0	0	0	0	0
Тек. вес мин. Пути (S)	4+2 = **6**	5	2	0	0

Гиперребра	A; B, C	A; D, B	A; C	B; D, E	C; D, E	C; E
К-во необр. конечных вершин (T)	0	0	0	0	0	0

Список вершин, ждущих обработки	C, A

ДОМАШНЕЕ ЗАДАНИЕ

- Обобщить на случай гиперграфов:
- - алгоритм Дейкстры
- - алгоритм A*
- - алгоритм вычисления суммы м-весов всех путей, проходящих через каждую из вершин графа