ОСНОВЫ ТЕОРИИ ГРАФОВ

Методические указания к проведению практических работ

Для студентов, обучающихся по направлению подготовки 230100.62 – «Информатика и вычислительная техника»

Составитель В. О. Гроппен

Владикавказ 2014

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего профессионального образования
"СЕВЕРО-КАВКАЗСКИЙ ГОРНО-МЕТАЛЛУРГИЧЕСКИЙ ИНСТИТУТ (ГОСУДАРСТВЕННЫЙ ТЕХНОЛОГИЧЕСКИЙ УНИВЕРСИТЕТ)"

Кафедра автоматизированной обработки информации

ОСНОВЫ ТЕОРИИ ГРАФОВ

Методические указания к проведению практических работ

Для студентов, обучающихся по направлению подготовки
230100.62 – «Информатика и вычислительная техника»

Составитель В. О. Гроппен

Допущено редакционно-издательским советом
Северо-Кавказского горно-металлургического
института (государственного технологического
университета)
Протокол заседания РИСа № 1 от 17.01.2014 г.

Владикавказ 2014

УДК 004.021
ББК 73
Г87

Рецензент
Кандидат технических наук,
доцент Северо-Кавказского горно-металлургического института
(государственного технического университета)
Будаева А. А.

Г87 **Основы теории графов:** Методические указания к проведению практических работ. Для студентов, обучающихся по направлению подготовки 230100.62 – "Информатика и вычислительная техника" / Сост. В. О. Гроппен; Северо-Кавказский горно-металлургический институт (государственный технологический университет). – Владикавказ: Северо-Кавказский горно-металлургический институт (государственный технологический университет). Изд-во «Терек», 2014. – с.

Методические указания предназначены для выполнения практических работ по курсу «Основы теории графов» для студентов по направлению подготовки 230100.62 «ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА» и содержат необходимые материалы по выполнению практических работ.

Подготовлено кафедрой «Автоматизированной обработки информации».

УДК 004.001
ББК 73

Редактор: *Иванченко Н. К.*
Компьютерная верстка: *Цишук Т. С.*

Подписано в печать 02.07.2014. Формат 60х84 $^{1}/_{16}$. Бумага офсетная.
Гарнитура «Таймс». Печать на ризографе. Усл. п.л. 2,67. Тираж 25 экз. Заказ № .
Северо-Кавказский горно-металлургический институт (государственный технологический университет). Издательство «Терек».
Отпечатано в отделе оперативной полиграфии СКГМИ (ГТУ).
362021, г. Владикавказ, ул. Николаева, 44.

Содержание

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 1

Цель: практическое усвоение алгоритма Прима.

Длительность занятия – 2 часа.

Решаемая задача: поиск минимальной базы ребер на взвешенном связном неориентированном графе**.**

Используемый алгоритм: алгоритм Прима**.**

Содержательная постановка задачи: на связном взвешенном неориентированном графе *G*(*X, U*) требуется выделить подмножество ребер $U' \subseteq U$ таких, что:

1. Граф *G*(*X, U'*) является связным.
2. Суммарный вес ребер подмножества *U'* является минимальным.

Определение: связным называется граф, между любой парой вершин которого существует маршрут.

Пример 1.1: минимальное связывающее подмножество ребер *U'* исходного графа *G*(*X, U*), изображенного на рисунке 1.1 слева, показано на том же рисунке справа (граф *G*(*X, U'*)).

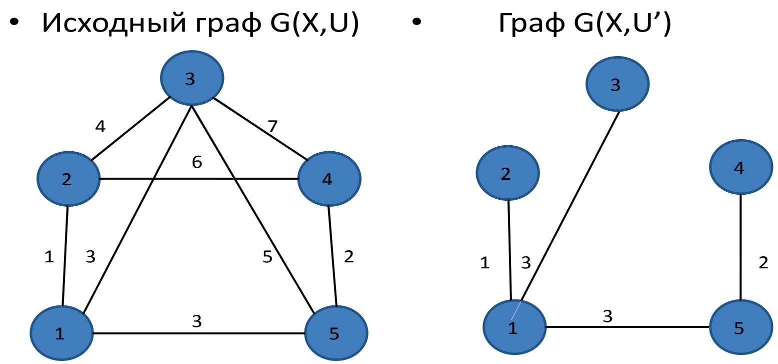


Рис. 1.1. Графы G(X,U) и G(X,U').

Формальная постановка этой задачи имеет вид:

$$\begin{cases} \sum\limits_i \sum\limits_j r(i,j)z(i,j) \to \min; \\ \forall p, \forall q \neq p : \sum\limits_d \prod\limits_{(i,j)\in L^d(p.q)} z(i,j) = 1; \\ \forall (i,j) \in U : z(i,j) = 1,0, \end{cases} \qquad (1.1)$$

где $L^{d}(p, d)$ – маршрут, соединяющий p-ю и q-ю вершины,

$z(i, j)$ – булева переменная, равная единице, если ребро (i, j) принадлежит подмножеству U' и равная нулю в противном случае.

$r(i, j)$ – вес ребра (i, j).

Для решения задачи (1.1) используется алгоритм Прима, описание которого приводится ниже.

Алгоритм Прима

Шаг 1. Выбирается произвольная i-я вершина.
Шаг 2. Выбирается инцидентное выбранной вершине ребро *(i,p)* с минимальным весом.
Шаг 3. Ребро *(i,p)* запоминается, а вершины i-я и p-я «стягиваются» в одну вершину.
Шаг 4. Вес «стянутого» ребра добавляется к ранее накопленной сумме.
Шаг 5. Если образовались параллельные ребра, то из них остается то, вес которого минимален, а остальные удаляются.
Шаг 6. Если все вершины графа стянуты в одну, то перейти к шагу 7, в противном случае – к шагу 1.
Шаг 7. Конец алгоритма. «Стянуты» искомые ребра.

Пример 1.2. На рисунке 1.2B – 1.2E показана последовательность преобразований исходного графа $G(X, U)$, изображенного на рисунке 1.2A, алгоритмом Прима. Граф $G(X, U')$ изображен на рисунке 1.2F.

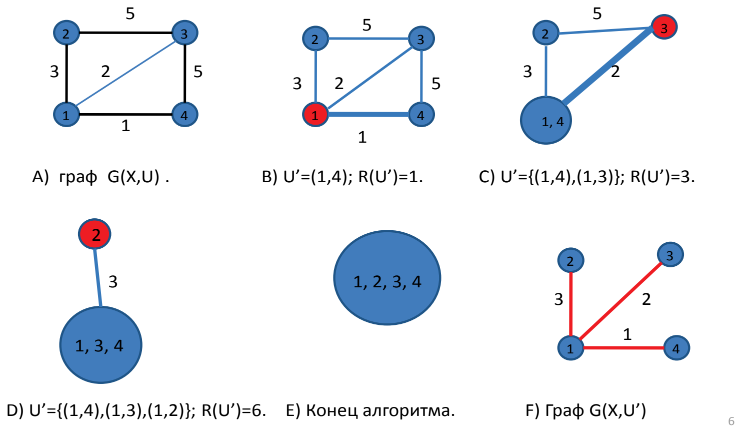


Рис. 1.2. Поиск минимального связывающего подмножества ребер (минимальной базы ребер) алгоритмом Прима.

Достоинства и недостатки алгоритма Прима

Достоинства:

1. Гарантия получения глобально оптимального решения.
2. Число итераций равно $|X| - 1$, где X – множество вершин.
3. Простота и наглядность.

Недостаток:

Алгоритм применим только к неориентированным графам.

Задания. Пользуясь матрицами μ_i, как матрицами инциденций для неориентированных, взвешенных и связных графов, определить на последних минимальные базы ребер.

$$\mu1 = \begin{array}{|c|c|c|c|c|}\hline 0&1&0&1&0\\\hline 1&0&0&0&1\\\hline 0&0&0&0&1\\\hline 1&0&0&0&0\\\hline 0&1&1&0&0\\\hline\end{array} \quad \mu2 = \begin{array}{|c|c|c|c|c|}\hline 0&1&1&1&1\\\hline 1&0&0&0&0\\\hline 1&0&0&0&0\\\hline 1&0&0&0&0\\\hline 1&0&0&0&0\\\hline\end{array} \quad \mu3 = \begin{array}{|c|c|c|c|c|}\hline 0&0&0&1&1\\\hline 0&0&0&0&1\\\hline 0&0&0&0&1\\\hline 1&0&0&0&0\\\hline 1&1&1&0&0\\\hline\end{array}$$

$$\mu4 = \begin{array}{|c|c|c|c|c|}\hline 0&1&3&1&1\\\hline 1&0&0&0&1\\\hline 3&0&0&0&1\\\hline 1&0&0&0&0\\\hline 1&1&1&0&0\\\hline\end{array} \quad \mu5 = \begin{array}{|c|c|c|c|c|}\hline 0&1&1&1&1\\\hline 1&0&0&0&1\\\hline 1&0&0&2&0\\\hline 1&0&2&0&3\\\hline 1&1&0&3&0\\\hline\end{array} \quad \mu6 = \begin{array}{|c|c|c|c|c|}\hline 0&4&0&1&1\\\hline 4&0&0&2&1\\\hline 0&0&0&0&1\\\hline 1&2&0&0&0\\\hline 1&1&1&0&0\\\hline\end{array}$$

$$\mu7 = \begin{array}{|c|c|c|c|c|}\hline 0&1&0&1&3\\\hline 1&0&0&2&1\\\hline 0&0&0&0&1\\\hline 1&2&0&0&5\\\hline 3&1&1&5&0\\\hline\end{array} \quad \mu8 = \begin{array}{|c|c|c|c|c|}\hline 0&1&1&1&1\\\hline 1&0&0&4&2\\\hline 1&0&0&0&3\\\hline 1&4&0&0&1\\\hline 1&2&3&1&0\\\hline\end{array} \quad \mu9 = \begin{array}{|c|c|c|c|c|}\hline 0&1&5&1&3\\\hline 1&0&6&2&1\\\hline 5&6&0&0&1\\\hline 1&2&0&0&4\\\hline 3&1&1&4&0\\\hline\end{array}$$

$$\mu10 = \begin{array}{|c|c|c|c|c|}\hline 0&1&0&1&3\\\hline 1&0&2&0&1\\\hline 0&2&0&0&1\\\hline 1&0&0&0&4\\\hline 3&1&1&4&0\\\hline\end{array} \quad \mu11 = \begin{array}{|c|c|c|c|c|}\hline 0&2&1&3&1\\\hline 2&0&0&0&0\\\hline 1&0&0&0&1\\\hline 3&0&0&0&4\\\hline 1&0&1&4&0\\\hline\end{array} \quad \mu12 = \begin{array}{|c|c|c|c|c|}\hline 0&0&0&1&2\\\hline 0&0&0&0&1\\\hline 0&0&0&0&3\\\hline 1&0&0&0&0\\\hline 2&1&3&0&0\\\hline\end{array}$$

$$\mu13 = \begin{array}{|c|c|c|c|c|}\hline 0&2&0&1&0\\\hline 2&0&0&0&3\\\hline 0&0&0&0&1\\\hline 1&0&0&0&4\\\hline 0&3&1&4&0\\\hline\end{array} \quad \mu14 = \begin{array}{|c|c|c|c|c|}\hline 0&4&1&1&2\\\hline 4&0&0&0&0\\\hline 1&0&0&1&3\\\hline 1&0&1&0&0\\\hline 2&0&3&0&0\\\hline\end{array} \quad \mu15 = \begin{array}{|c|c|c|c|c|}\hline 0&0&0&1&2\\\hline 0&0&5&0&1\\\hline 0&5&0&0&1\\\hline 1&0&0&0&0\\\hline 2&1&1&0&0\\\hline\end{array}$$

$$\mu16 = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 0 & 1 & 0 \\ \hline 1 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 1 & 1 & 0 & 0 \\ \hline \end{array} \quad \mu17 = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 0 & 0 & 0 \\ \hline \end{array} \quad \mu18 = \begin{array}{|c|c|c|c|c|}\hline 0 & 0 & 1 & 1 & 1 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 1 & 1 & 1 & 0 & 0 \\ \hline \end{array}$$

$$\mu19 = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 0 & 3 & 0 \\ \hline 1 & 0 & 0 & 0 & 2 \\ \hline 0 & 0 & 0 & 0 & 1 \\ \hline 3 & 0 & 0 & 0 & 0 \\ \hline 0 & 2 & 1 & 0 & 0 \\ \hline \end{array} \quad \mu20 = \begin{array}{|c|c|c|c|c|}\hline 0 & 3 & 1 & 2 & 1 \\ \hline 3 & 0 & 0 & 0 & 0 \\ \hline 1 & 0 & 0 & 4 & 0 \\ \hline 2 & 0 & 4 & 0 & 5 \\ \hline 1 & 0 & 0 & 5 & 0 \\ \hline \end{array} \quad \mu21 = \begin{array}{|c|c|c|c|c|}\hline 0 & 4 & 0 & 1 & 2 \\ \hline 4 & 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 & 3 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 2 & 1 & 3 & 0 & 0 \\ \hline \end{array}$$

$$\mu22 = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 4 & 1 & 0 \\ \hline 1 & 0 & 0 & 2 & 1 \\ \hline 4 & 0 & 0 & 0 & 3 \\ \hline 1 & 2 & 0 & 0 & 0 \\ \hline 0 & 1 & 3 & 0 & 0 \\ \hline \end{array} \quad \mu23 = \begin{array}{|c|c|c|c|c|}\hline 0 & 1 & 2 & 1 & 3 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 2 & 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 0 & 0 & 4 \\ \hline 3 & 0 & 1 & 4 & 0 \\ \hline \end{array} \quad \mu24 = \begin{array}{|c|c|c|c|c|}\hline 0 & 2 & 0 & 1 & 4 \\ \hline 2 & 0 & 5 & 0 & 1 \\ \hline 0 & 5 & 0 & 0 & 3 \\ \hline 1 & 0 & 0 & 0 & 0 \\ \hline 4 & 1 & 3 & 0 & 0 \\ \hline \end{array}$$

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 2

Цель: практическое усвоение метода поиска минимаксного остова неориентированного взвешенного графа.

Длительность занятия – 2 часа.

Используемый алгоритм: дихотомия.

Задача о минимаксном остове взвешенного графа

Содержательная постановка задачи: на взвешенном неориентированном графе $G(X, U)$ требуется выделить подмножество ребер $U' \subseteq U$ таких, что:

1. Отношения достижимости вершин графов $G(X, U')$ и $G(X, U)$ совпадают.

2. Максимальный вес ребра подмножества U' является минимальным.

Пример 2.1: минимаксный остов (подмножество ребер U') исходного графа $G(X, U)$, изображенного на рисунке 2.1 слева, показан на том же рисунке справа (граф $G(X, U')$).

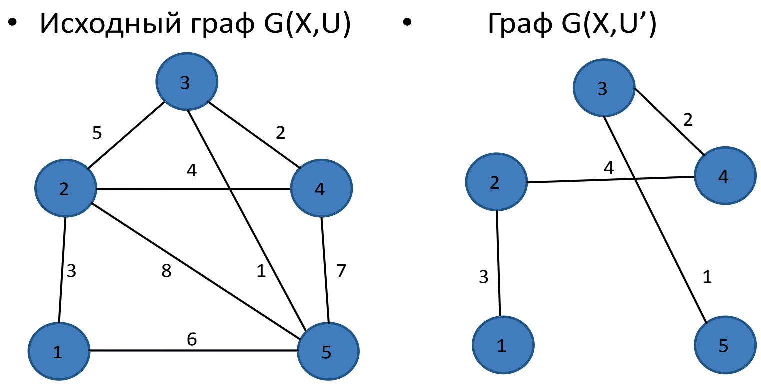


Рис. 2.1. Исходный и оптимальный графы.

Формальная постановка этой задачи имеет вид:

$$\begin{cases} \max\limits_{i} \max\limits_{j} r(i,j) z(i,j) \to \min; \\ \forall p, \forall q \neq p : \sum\limits_{d} \prod\limits_{(i,j) \in L_1^d(p.q)} z(i,j) = \sum\limits_{d} \prod\limits_{(i,j) \in L_2^d(p.q)} z(i,j); \\ \forall (i,j) \in U : z(i,j) = 1,0, \end{cases} \tag{2.1}$$

где $L_1^d(p,d)$ – d-й маршрут, соединяющий p-ю и q-ю вершины графа $G(X, U)$,

$L_2^d(p,d)$ – d-й маршрут, соединяющий p-ю и q-ю вершины графа $G(X, U')$,

$z(i, j)$ – булева переменная, равная единице, если ребро (i, j) принадлежит подмножеству U' и равная нулю в противном случае.

$r(i, j)$ – вес ребра (i, j).

Одним из способов решения этой задачи является использование дихотомии. Ниже приводится описание алгоритма, в основе которого лежит эта операция.

Алгоритм 2.1

Шаг 1. A = 1

Шаг 2. B = $|U|$.

Шаг 3. Ищется перестановка ребер $\pi = \{(i,j)_1, (i,j)_2, (i,j)_3, \ldots\}$ такая, что справедливо неравенство: $\forall 1 \le k \le |U|-1 : r(i,j)_k \le r(i,j)_{k+1}$.

Шаг. 4. C = целой части от {(A + B)/2}.

Шаг 5. Из исходного графа удаляются все ребра, индекс k которых в упорядочении π превышает C. Подмножество оставшихся ребер обозн ачается U'.

Шаг 6. Если для полученного и исходного графа совпадают условия достижимости вершин:

$$\forall p, \forall q \ne p : \sum_d \prod_{(i,j)\in L_1^d(p.q)} z(i,j) = \sum_d \prod_{(i,j)\in L_2^d(p.q)} z(i,j),$$

то перейти к шагу 7, в противном случае – к шагу 9.

Шаг 7. Если B – A < 2, то перейти к шагу 10, в противном случае – к шагу 8.

Шаг 8. B = C, перейти к шагу 4.

Шаг 9. A = C, перейти к шагу 4.

Шаг 10. Конец алгоритма. Значение целевой функции равно весу ребра, стоящего на B-м месте в перестановке π.

Пример 2.1. Пользуясь приведенным выше алгоритмом, определить минимаксный остов графа, изображенного на рис. 2.1 слева.

Решение.

1. A=1.
2. B=8.
3. π = {(3,5), (3,4), (1,2), (2,4), (2,3), (1,5), (4,5), (2,5)}.
4. C = entire{(A + B)/2}=4.
5. U'={(3,5), (3,4), (1,2), (2,4)}.

6. Граф G(X,U’) изображен на рисунке 2.1 справа, отношения достижимости его вершин совпадают с отношениями достижимости вершин графа G(X,U).

7. Так как B – A > 2, то B = C = 4.

8. C = 2.

9. U’={(3,5), (3,4)}.

10. Очевидно, что отношения достижимости вершин графов G(X,U’) и G(X,U) не совпадают.

11. A = C = 2.

12. Так как разность B – A = 2, определяется новое значение C = 3.

13. U’={(3,5), (3,4), (1,2)}.

14. Очевидно, что отношения достижимости вершин графов G(X,U’) и G(X,U) вновь не совпадают.

15. A = C=3.

16. Так как B – A < 2, то алгоритм закончен. Оптимальное подмножество U’={(3,5), (3,4), (1,2), (2,4)}, оптимальное значение целевой функции равно четырем, граф G(X,U’) изображен на рисунке 2.1 справа.

Достоинства и недостатки алгоритма 2.1

Достоинства:

1. Гарантия получения глобально оптимального решения.
2. Сравнительно высокое быстродействие.
3. Легкость программной реализации.
4. Возможность использовать алгоритм для орграфов, изменив шаг 6.

5. Недостатки:
6. Перебор, реализуемый на шаге 6, понижает быстродействие алгоритма.
7. После того, как оптимальное решение найдено, алгоритм продолжает поиск, чтобы убедиться в этом.

Задания. Пользуясь матрицами μ_i, как матрицами инциденций для неориентированных, взвешенных и связных графов, определить на последних <u>минимаксные</u> остовы ребер.

μ1 =

0	1	0	1	0
1	0	0	0	1
0	0	0	0	1
1	0	0	0	0
0	1	1	0	0

μ2 =

0	1	1	1	1
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0
1	0	0	0	0

μ3 =

0	0	0	1	1
0	0	0	0	1
0	0	0	0	1
1	0	0	0	0
1	1	1	0	0

$\mu_4 =$

0	1	3	1	1
1	0	0	0	1
3	0	0	0	1
1	0	0	0	0
1	1	1	0	0

$\mu_5 =$

0	1	1	1	1
1	0	0	0	1
1	0	0	2	0
1	0	2	0	3
1	1	0	3	0

$\mu_6 =$

0	4	0	1	1
4	0	0	2	1
0	0	0	0	1
1	2	0	0	0
1	1	1	0	0

Задание 3

$\mu_7 =$

0	1	0	1	3
1	0	0	2	1
0	0	0	0	1
1	2	0	0	5
3	1	1	5	0

$\mu_8 =$

0	1	1	1	1
1	0	0	4	2
1	0	0	0	3
1	4	0	0	1
1	2	3	1	0

$\mu_9 =$

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

Задание 4

$\mu_{10} =$

0	1	0	1	3
1	0	2	0	1
0	2	0	0	1
1	0	0	0	4
3	1	1	4	0

$\mu_{11} =$

0	2	1	3	1
2	0	0	0	0
1	0	0	0	1
3	0	0	0	4
1	0	1	4	0

$\mu_{12} =$

0	0	0	1	2
0	0	0	0	1
0	0	0	0	3
1	0	0	0	0
2	1	3	0	0

Задание 5

$\mu_{13} =$

0	2	0	1	0
2	0	0	0	3
0	0	0	0	1
1	0	0	0	4
0	3	1	4	0

$\mu_{14} =$

0	4	1	1	2
4	0	0	0	0
1	0	0	1	3
1	0	1	0	0
2	0	3	0	0

$\mu_{15} =$

0	0	0	1	2
0	0	5	0	1
0	5	0	0	1
1	0	0	0	0
2	1	1	0	0

$\mu_{16} =$

0	1	0	1	0
1	0	0	0	1
0	0	0	0	1
1	0	0	0	0
0	1	1	0	0

$\mu_{17} =$

0	1	1	1	1
1	0	0	0	1
1	0	0	0	0
1	0	0	0	0
1	1	0	0	0

$\mu_{18} =$

0	0	1	1	1
0	0	0	0	1
1	0	0	0	1
1	0	0	0	0
1	1	1	0	0

$\mu_{19} =$

0	1	0	3	0
1	0	0	0	2
0	0	0	0	1
3	0	0	0	0
0	2	1	0	0

$\mu_{20} =$

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

$\mu_{21} =$

0	4	0	1	2
4	0	0	0	1
0	0	0	0	3
1	0	0	0	0
2	1	3	0	0

$\mu_{22} =$

0	1	4	1	0
1	0	0	2	1
4	0	0	0	3
1	2	0	0	0
0	1	3	0	0

$\mu_{23} =$

0	1	2	1	3
1	0	0	0	0
2	0	0	0	1
1	0	0	0	4
3	0	1	4	0

$\mu_{24} =$

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

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 3

Цель: практическое усвоение метода поиска максимального паросочетания с минимальным весом на бихроматическом взвешенном графе.

Длительность занятия – 2 часа.

Задача о поиске максимального паросочетания с минимальным весом бихроматического взвешенного графа

Бихроматические графы

Будем считать, что задан неориентированный двудольный граф $G(X, U)$, если множество его вершин X состоит из двух непересекающихся подмножеств, а любое ребро множества U соединяет вершины, принадлежащие различным подмножествам. Формально это может быть описано следующим образом:

X' – «левое» подмножество вершин;

X'' – «правое» подмножество вершин ($X' U X'' = X$);

$\forall x_i \in X', \forall x_j \in X' : (i, j) \notin U$;

$\forall x_i \in X'', \forall x_j \in X'' : (i, j) \notin U$.

Содержательная постановка задачи о максимальном паросочетании

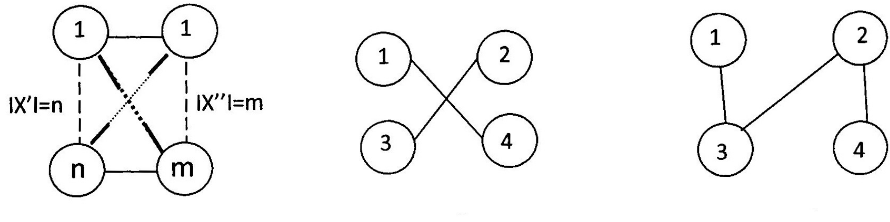
На множестве ребер U графа $G(X, \boldsymbol{U})$ требуется выделить подмножество $U' \subseteq U$, такое, что:

– существует не более одного ребра, принадлежащего U' и инцидентного каждой вершине подмножества X';

– существует не более одного ребра, принадлежащего U' и инцидентного каждой вершине подмножества X'';

– мощность множества U' максимальна.

Определение**:** Подмножество U' ребер называется ***паросочетанием***, если любые два ребра этого подмножества не имеют общей вершины.



а) *б*) *в*)

Рис. 3.1. Иллюстрации: *а*) бихроматический граф; *б*) паросочетание; *в*) граф не является ни бихроматическим, ни паросочетанием.

Пусть булева переменная $y(i, j)$ равна единице, если ребро (i, j) принадлежит подмножеству U', и равна нулю в противном случае. Тогда формальная постановка задачи поиска максимального паросочетания имеет вид:

$$\begin{cases} \sum\limits_{i=1}^{n} \sum\limits_{j=1}^{n} y(i,j) \to \max; \quad \textit{- целевая функция ;} \\ \sum\limits_{i=1}^{n} y(i,j) \le 1, \quad j = 1,2,...,n; \\ \sum\limits_{j=1}^{n} y(i,j) \le 1, \quad i = 1,2,...,n; \\ y(i,j) = 1,0; \quad i = 1,2,...,n; \quad j = 1,2,...,n\ , \end{cases} \tag{3.1}$$

где:

$$\begin{cases} y(i, j) = 1 \Rightarrow (i, j) \in U'; \\ y(i, j) = 0 \Rightarrow (i, j) \notin U'. \end{cases}$$

Поиск максимального паросочетания является частным случаем целого класса задач, который получил название «задачи о назначениях».

Задача о назначениях с аддитивным функционалом цели

Заданы n работ и n рабочих, причем известна стоимость $r(i, j)$ выполнения i-м рабочим j-й работы. Требуется распределить работы между рабочими таким образом, чтобы:

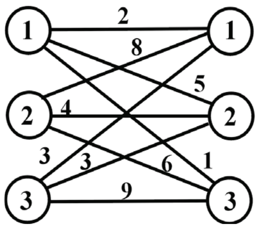
1. Все работы были выполнены;
2. Все рабочие были заняты;
3. Суммарные затраты на выполнение всех работ были минимальны.

Формальная постановка этой задачи имеет вид:

$$
\begin{cases}
\sum\limits_{i=1}^{n}\ \sum\limits_{j=1}^{n} r(i,j)\, y(i,j) \to \min; & \text{- целевая функция - минимизация затрат;}\\
\sum\limits_{i=1}^{n} y(i,j) = 1, \quad j = 1,2,\ldots,n; & \text{- условие выполнения всех работ;}\\
\sum\limits_{j=1}^{n} y(i,j) = 1, \quad i = 1,2,\ldots,n; & \text{- условие загруженности всех рабочих;}\\
y(i,j) = 1,0; \quad i = 1,2,\ldots,n; \quad j = 1,2,\ldots,n & \text{– дискретность переменных}
\end{cases}
\tag{3.2}
$$

Примечание: если i-й рабочий не может делать j-ю работу, то *r(i,j)* = ∞.

Форма представления данных – бихроматический граф (рис.3.2*а*) либо таблица, строки которой соответствуют вершинам подмножества *X* ’, а столбцы – вершинам подмножества *X* ” (рис. 3.2*б*).



а)

	1	2	3
1	**2**	**5**	**1**
2	**8**	**4**	**6**
3	**3**	**3**	**9**

б)

Рис. 3.3. Примеры представления исходных данных задачи 3.2.

Алгоритм 3.1

Шаг 1. $i = 1$

Шаг 2. В i-й строке матрицы M выбирается элемент, вес которого равен $Q = \min M(i, j)$ и уменьшаем вес каждого элемента этой строки на Q.

Шаг 3. $i = i + 1$

Шаг 4. Если $i > n$, то перейти к Шагу 5, нет к Шагу 2.

Шаг 5. $j = 1$

Шаг 6. В j-ом столбце матрицы M выбирается элемент, вес которого равен $D = \min M(i, j)$.

Шаг 7. Вес каждого элемента j-го столбца уменьшается на величину D.

Шаг 8. $j = j + 1$.

Шаг 11. Если $L = n$, то перейти к Шагу 14, в противном случае – к Шагу 12.

Шаг 12. На множестве неперечеркнутых элементов матрицы M выбирается тот, вес которого минимален и равен W.

Шаг 13. Вес неперечеркнутых элементов матрицы уменьшаем на W, а перечеркнутых дважды – увеличиваем на W. Перейти к Шагу 8.

Шаг 14. Конец алгоритма. На множестве нулей полученной матрицы есть оптимальное назначение.

Задания. Пользуясь (5 х 5) матрицами M, как матрицами инциденций для неориентированных, взвешенных и бихроматических графов, определить на последних максимальное паросочетание с минимальным суммарным весом ребер.

Задание 1

M =

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

Задание 2

M =

0	0	1	1	5
1	0	1	1	1
0	4	0	1	1
0	0	3	0	1
2	0	0	0	7

Задание 3

M =

0	0	1	1	4
0	0	1	0	1
0	0	0	1	1
0	1	5	0	1
1	0	0	3	0

Задание 4

M =

0	1	1	2	1
0	0	1	1	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

M =

0	1	1	1	0
4	0	1	1	1
0	0	0	1	1
0	0	2	0	1
0	3	0	4	0

Задание 6

M =

0	2	1	1	1
1	0	1	1	1
0	0	0	1	1
0	2	0	0	1
3	0	0	1	0

Задание 7

M =

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

Задание 8

M =

0	0	1	1	3
1	0	1	1	1
0	2	0	1	1
0	4	0	0	1
1	0	3	0	0

Задание 9

M =

0	1	1	1	0
3	0	1	1	2
0	0	4	0	1
0	5	0	0	1
7	1	0	0	0

Задание 10

M =

0	0	1	1	1
1	0	2	1	1
0	1	0	1	1
0	2	0	0	1
5	0	0	3	0

Задание 11

M =

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
4	2	0	0	1
0	1	3	0	0

Задание 12

M =

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

Задание 13

M =

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	8

Задание 14

M =

0	0	1	1	3
1	8	1	2	1
0	0	5	1	1
0	1	0	0	1
1	0	0	4	0

Задание 15

M =

2	1	0	1	1
0	0	1	1	1
1	0	0	0	1
0	5	1	0	1
1	0	1	0	7

Задание 16

M =

6	2	1	0	1
1	2	0	1	1
0	1	3	1	1
1	1	0	0	1
1	0	0	3	0

Задание 17

M =

4	1	1	1	2
0	7	1	1	1
0	0	0	1	3
0	5	0	2	1
1	0	1	0	0

Задание 18

M =

0	3	1	1	1
1	0	0	1	1
2	1	9	1	1
0	0	0	5	4
1	0	0	1	3

Задание 19

M =

0	1	1	1	0
0	0	1	1	3
4	0	0	2	1
0	0	1	0	1
9	1	0	0	2

Задание 20

M =

0	0	1	2	1
1	0	1	1	1
0	9	5	1	1
1	0	0	0	1
1	5	3	2	0

Задание 21

M =

9	1	1	1	0
4	0	1	1	0
0	3	0	1	1
0	1	0	7	1
0	1	0	2	8

Задание 22

M =

0	0	1	1	7
1	9	1	4	1
0	0	7	1	1
0	1	2	0	1
1	0	1	0	2

Задание 23

M =

0	1	1	1	2
2	0	1	1	1
0	0	0	1	1
0	3	0	7	6
1	0	4	0	8

Задание 24

M =

0	3	1	1	5
1	5	1	1	1
3	0	0	1	1
0	0	0	9	1
2	0	8	4	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 4

Цель: практическое усвоение метода поиска максимального паросочетания бихроматического взвешенного графа.

Длительность занятия – 2 часа.

Задача о поиске максимального паросочетания с минимаксным весом бихроматического взвешенного графа

Эта задача называется также «задачей о назначении на узкие места» из-за одной из ее интерпретаций: надо расставить n рабочих на n мест на конвейере, если известно время выполнения «своей» операции $r(i, j)$ каждого i-го рабочего на каждом j-м рабочем месте. Очевидно, что производительность конвейера определяется самым «медлительным» рабочим. Т.о. следует так расставить рабочих, чтобы время выполнения операции самым медлительным рабочим на конвейере было минимальным. Формальная постановка задачи имеет вид:

$$\begin{cases} \max\limits_{i} \max\limits_{j}\ r\ (i, j)\ y(i, j) \to \min; \ - \text{целевая функция}; \\ \sum\limits_{i=1}^{n} y(i, j) = 1; \ \ j = 1,2,\ldots,n; \ \ - \text{каждое рабочее место должно быть заполнено}; \\ \sum\limits_{j=1}^{n} y(i, j) = 1; \ \ i = 1,2,\ldots,n; \ \ - \text{каждый рабочий должен быть занят}; \\ y(i, j) = 1,0; \ \ i = 1,2,\ldots,n; \ \ j = 1,2,\ldots,n. \end{cases} \tag{4.1}$$

Ниже приводится описание алгоритма решения задачи (4.1) и примера, иллюстрирующего его работу. При этом полагаем, что величины $r(i, j)$, определяющие времена выполнения операций для каждого i-го рабочего на каждом j-м рабочем месте, определяются графом $G(X,U)$, $X = X_1 U X_2$, $|X_1| = |X_2| = n$ и соответствующей ему матрицей M.

Алгоритм 4.1

Шаг 1. С = ∞.

Шаг 2. Из графа удаляются все ребра и ищется их упорядочение π такое, что $r(i, j)_k \le r(i, j)_{k+1}$.

Шаг 3. $q = 1$.

Шаг 4. В граф возвращаются первые q ребер упорядочения π.

Шаг 5. Алгоритмом 3.1 на полученной матрице M ищется оптимальное решение задачи (4.1).

Шаг 6. Если полученное на предыдущем шаге значение целевой функции равно ∞, то перейти к шагу 8, в противном случае – к шагу 7.

Шаг 7. На множестве оптимальных назначений, полученных на последней итерации алгоритмом 3.1, выбирается такое, которому соответствует минимальное значение, обозначаемое символом S. Перейти к шагу 10.

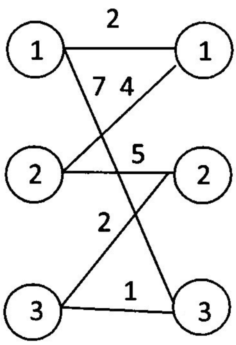
Шаг 8. $q = q + 1$. Если величина q превышает число ребер графа, то перейти к шагу 9, в противном случае перейти к шагу 4.

Шаг 9. $S = C$.

Шаг 10. Конец алгоритма. Величина S равна значению целевой функции системы (4.1) при оптимальных значениях булевых переменных.

Пример, иллюстрирующий работу алгоритма 1.4.7, приводится ниже.

Пример 4.1. Решить задачу (4.1) применительно к графу $G(X, U)$, изображенному ниже на рисунке 4.1.



а)

2	0	7
4	5	0
0	2	1

б)

Рис. 4.1. Бихроматический граф $G(X, U)$ и его матрица M.

1. $C = \infty$.
2. Генерируется перестановка $\pi = \{(3, 3); (1, 1); (3, 2); (2, 2); (1, 3)\}$.
3. На первой итерации $S = \infty$, ($q = 1$) получаем матрицу:

∞	∞	∞
∞	∞	∞
∞	∞	1

4. На второй итерации $S = \infty$, матрица M преобразуется к виду:

2	∞	∞
∞	∞	∞
∞	∞	1

5. После третьей итерации $S = \infty$, M принимает вид:

2	∞	∞
∞	∞	∞
∞	2	1

6. После четвертой итерации M принимает вид:

2	∞	∞
∞	5	∞
∞	2	1

7. Матрица М, полученная на пятой итерации:

2	∞	7
4	5	∞
∞	2	1

8. Решение, полученное алгоритмом 3.1:

2	∞	∞
4	5	∞
∞	2	1

9. $S = 5$. Конец алгоритма.

Задания. Пользуясь (5 x 5) матрицами M, как матрицами инциденций для неориентированных, взвешенных и бихроматических графов, определить на последних максимальное паросочетание с минимаксным весом ребер.

Задание 1

$M =$

0	1	1	1	0
2	0	1	1	1
0	7	0	1	4
4	0	3	0	1
0	0	2	5	0

Задание 2

$M =$

0	0	1	1	5
1	0	1	1	1
0	4	0	1	1
0	0	3	0	1
2	0	0	0	7

Задание 3

$M =$

3	0	1	1	4
0	9	1	0	1
7	0	0	1	1
0	1	5	0	1
1	0	0	3	0

Задание 4

$M =$

0	1	1	2	1
8	0	1	1	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

$M =$

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

Задание 6

$M =$

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

Задание 7

$M =$

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

Задание 8

$M =$

0	0	9	1	3
1	0	1	5	1
0	2	0	1	8
0	4	0	12	1
1	0	3	0	0

Задание 9

$M =$

0	1	1	1	0
3	0	1	1	2
0	0	4	0	1
0	5	0	0	1
7	1	0	0	0

Задание 10

$M =$

0	0	1	1	1
1	0	2	1	1
0	1	0	1	1
0	2	0	0	1
5	0	0	3	0

Задание 11

$M =$

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
4	2	0	0	1
0	1	3	0	0

Задание 12

$M =$

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

Задание 13

$M =$

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	8

Задание 14

$M =$

0	0	1	1	3
1	8	1	2	1
0	0	5	1	1
0	1	0	0	1
1	0	0	4	0

Задание 15

$M =$

2	1	0	8	1
0	0	1	1	1
1	0	0	0	9
0	5	1	0	1
1	0	1	0	7

Задание 16

$M =$

6	2	1	0	1
1	2	0	1	1
0	1	3	1	1
1	1	0	0	1
1	0	0	3	0

Задание 17

$M =$

4	1	1	1	2
0	7	1	4	1
0	0	0	1	3
0	5	0	2	1
1	0	1	0	0

Задание 18

$M =$

0	3	1	7	5
1	0	0	1	8
2	1	9	1	1
0	0	0	5	4
3	0	0	1	3

Задание 19

$M =$

0	1	1	1	0
0	0	1	1	3
4	0	0	2	1
0	0	1	0	1
9	1	0	0	2

Задание 20

$M =$

0	0	1	2	1
1	0	4	1	7
0	9	5	1	8
1	0	0	0	9
1	5	3	2	0

Задание 21

$M =$

9	1	1	1	0
4	0	1	1	0
0	3	0	1	1
0	1	0	7	1
0	1	0	2	8

Задание 22

$M =$

0	0	2	1	7
1	9	1	4	3
0	0	7	1	4
0	1	2	0	5
1	0	6	0	2

Задание 23

$M =$

0	1	12	1	2
2	0	9	1	1
0	0	0	10	5
0	3	0	7	6
11	0	4	0	8

Задание 24

$M =$

0	3	1	1	5
10	5	11	1	1
3	0	6	1	12
0	7	0	9	1
2	0	8	4	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 5

Цель: практическое усвоение метода поиска кратчайшего пути на взвешенном орграфе.

Длительность занятия – 2 часа.

Задача о поиске кратчайшего пути из i-й вершины в j-ю на взвешенном орграфе методом потенциалов.

Формальная постановка задачи поиска кратчайшего пути из s-й вершины в t-ю:

$$\sum_i \sum_j r(i,j)z(i,j) \to \min;$$

$$\sum_i z(s,i) = 1;$$

$$\sum_i z(i,t) = 1;$$

$$\forall x_k \in X \setminus (x_s \cup x_t) : \sum_i z(i,k) = \sum_j z(k,j);$$

$$\forall i, \forall j : z(i,j) = 1,0.$$

Алгоритм, реализующий метод потенциалов

Шаг 1. Вершине x_s присваивается потенциал $P(s)=0$.

Шаг 2. Всем вершинам множества $X\setminus x_s$ присвоить потенциал, равный ∞.

Шаг 3. Каждой q-й вершине множества $X\setminus x_s$ присваивается потенциал $P(q)$:

$$\forall x_q \in X \setminus x_s : P(q) = \min\{P(q); \min_i [P(i) + r(i,q)]\}.$$

Шаг 4. Если потенциал хотя бы одной вершины изменился, то перейти к шагу 3, в противном случае – к шагу 5.

Шаг 5. Конец алгоритма. Потенциал t-й вершины равен кратчайшему пути в нее из s-й вершины.

Достоинства и недостатки алгоритма;

Достоинства:

1. Метод потенциалов гарантирует определение кратчайших путей из выбранной вершины во все остальные.
2. Исключается необходимость перебора всех путей.

3. Высокое быстродействие.

4. Легкая программная реализация.

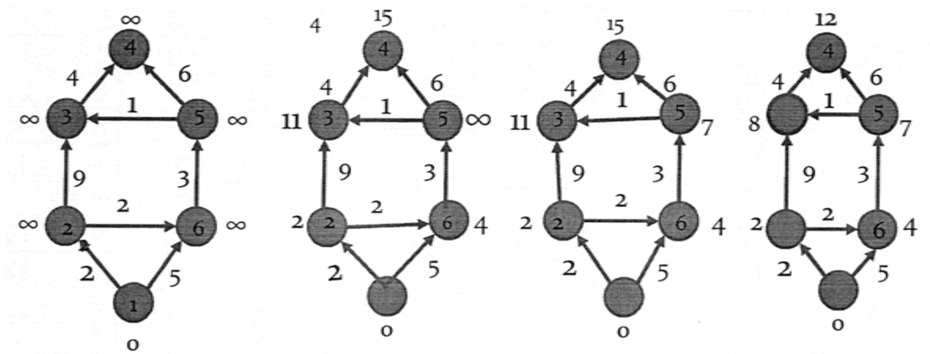
5. Универсальность: метод применим к ориентированным и неориентированным графам.

Недостатки:

Вес каждой дуги должен быть положительным.

Пример 5.1.

Поиск длины кратчайшего пути из 1-й вершины в 4-ю.



а) *б*) *в*) *г*)

Порядок расстановки потенциалов на каждой интерации – по часовой стрелке.

Задания. Пользуясь (5 x 5) матрицами *M*, как матрицами инциденций для ориентированных и взвешенных графов, определить на последних кратчайшие пути из 1-й в 5-ю вершину методом потенциалов.

Задание 1

$M =$

0	1	1	1	10
2	0	1	1	12
0	7	0	1	4
4	0	3	0	1
0	0	2	5	0

Задание 2

$M =$

0	0	1	1	5
1	0	1	1	11
0	4	0	1	8
0	0	3	0	1
2	0	0	0	0

Задание 3

$M =$

0	0	1	1	4
0	0	1	0	5
7	0	0	1	6
0	1	5	0	3
1	0	0	3	0

Задание 4

$M =$

0	1	1	2	17
8	0	1	1	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

$M =$

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

Задание 6

$M =$

0	2	1	1	14
1	0	1	1	4
0	0	0	6	8
0	2	0	0	5
3	0	0	1	0

Задание 7

$M =$

0	7	1	2	10
2	0	1	0	1
0	0	0	1	6
1	4	0	0	1
5	0	0	3	0

Задание 8

$M =$

0	0	9	5	13
1	0	1	5	1
0	2	0	1	8
0	4	0	0	4
1	0	3	0	0

Задание 9

$M =$

0	9	1	5	10
3	0	1	1	2
0	0	0	0	10
0	5	0	0	5
7	1	0	0	0

Задание 10

$M =$

0	0	9	10	11
1	0	2	12	13
0	1	0	1	1
0	2	0	0	7
5	0	0	3	0

Задание 11

$M =$

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
4	2	0	0	1
0	1	3	0	0

Задание 12

$M =$

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

Задание 13

$M =$

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	0

Задание 14

$M =$

0	0	1	9	0
1	0	1	2	10
0	0	0	1	17
0	1	0	0	1
1	0	0	4	0

Задание 15

$M =$

0	1	0	8	0
0	0	1	1	12
1	0	0	0	9
0	5	1	0	1
1	0	1	0	0

Задание 16

$M =$

0	2	1	0	0
1	0	0	1	11
0	1	0	1	1
1	1	0	0	9
1	0	0	3	0

Задание 17

$M =$

0	1	7	8	12
0	0	1	4	9
0	0	0	1	3
0	5	0	0	4
1	0	1	0	0

Задание 18

$M =$

0	3	4	7	15
1	0	0	1	8
2	1	0	1	1
0	0	0	0	4
3	0	0	1	0

Задание 19

$M =$

0	12	5	6	0
0	0	1	1	3
4	0	0	2	7
0	0	1	0	9
9	1	0	0	0

Задание 20

$M =$

0	0	15	2	11
1	0	4	1	7
0	9	0	1	8
1	0	0	0	9
1	5	3	2	0

Задание 21

$M =$

0	1	5	4	0
4	0	1	1	0
0	3	0	1	8
0	1	0	0	6
0	1	0	2	0

Задание 22

$M =$

0	0	2	1	7
1	0	1	4	3
0	0	0	1	4
0	1	2	0	5
1	0	6	0	0

Задание 23

$M =$

0	1	12	5	20
2	0	9	1	12
0	0	0	10	5
0	3	0	0	6
11	0	4	0	0

Задание 24

$M =$

0	3	4	7	5
10	0	11	1	3
3	0	0	1	12
0	7	0	0	6
2	0	8	4	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 6

Цель: практическое усвоение метода поиска минимальной базы дуг на взвешенном орграфе.

Длительность занятия – 2 часа.

Задача о поиске минимальной базы дуг на взвешенном орграфе

***Определение 1*:** Базой дуг ориентированного графа $G(X, U)$ с матрицей достижимости вершин «M» называется такое подмножество дуг U' множества U, что:

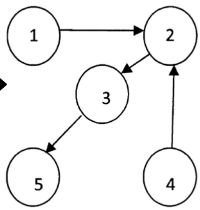
⦿ граф $G(X, U')$ обладает такой же матрицей достижимости вершин M', что и исходный граф $G(X, U)$.

⦿ Удаление любой дуги, принадлежащей базе U', изменяет условия достижимости вершин.

Пример 6.1

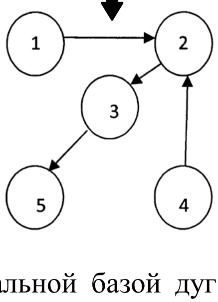
0	**1**	**1**	**0**	**1**
0	**0**	**1**	**0**	**0**
0	**0**	**0**	**0**	**1**
0	**1**	**1**	**0**	**1**
0	**0**	**0**	**0**	**0**

Матрица смежности вершин



1	**1**	**1**	**0**	**1**
0	**1**	**1**	**0**	**1**
0	**0**	**1**	**0**	**1**
0	**1**	**1**	**1**	**1**
0	**0**	**0**	**0**	**1**

Матрица достижимости вершин



База дуг

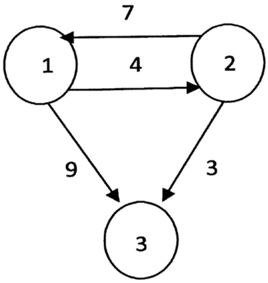
***Определение 2*:** Минимальной базой дуг взвешенного ориентированного графа $G(X, U)$ с матрицей достижимости вершин «M» называется такое подмножество дуг U' множества U, что:

⦿ граф $G(X, U')$ обладает такой же матрицей достижимости вершин M', что и исходный граф $G(X, U)$;

⦿ суммарный вес дуг подмножества U' минимален.

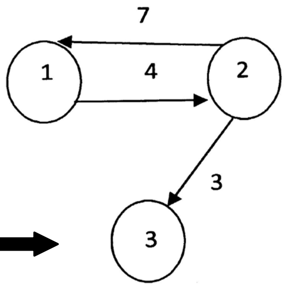
Поиск минимальной базы дуг перебором

Пример 6.2



Исходный граф

Суграф с минимальной базой дуг



№	Z(1,3)	Z(2,3)	Z(1,2)	Z(2,1)	R
1	**0**	**0**	**0**	**1**	∞
2	**0**	**0**	**1**	**0**	∞
3	**0**	**0**	**1**	**1**	∞
4	**0**	**1**	**0**	**0**	∞
5	**0**	**1**	**0**	**1**	∞
6	**0**	**1**	**1**	**0**	∞
7	**0**	**1**	**1**	**1**	**14**

Поиск минимальной базы дуг перебором.

Задания. Пользуясь (5 x 5) матрицами M, как матрицами инциденций для ориентированных и взвешенных графов, определить на последних минимальные базы дуг перебором.

Задание 1

$M =$

0	1	1	1	10
0	0	1	1	12
0	7	0	1	4
0	0	3	0	1
0	0	2	5	0

Задание 2

$M =$

0	0	1	1	5
1	0	1	1	11
0	0	0	1	8
0	0	3	0	1
0	0	0	0	0

Задание 3

$M =$

0	0	0	1	4
0	0	0	0	5
7	0	0	1	6
0	1	0	0	3
1	0	0	3	0

Задание 4

$M =$

0	1	1	0	17
8	0	1	0	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

$M =$

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

Задание 6

$M =$

0	2	1	1	14
0	0	1	1	4
0	0	0	6	8
0	2	0	0	5
0	0	0	1	0

Задание 7

$M =$

0	7	1	2	10
2	0	1	0	1
0	0	0	1	6
1	4	0	0	1
5	0	0	3	0

Задание 8

$M =$

0	0	9	5	13
1	0	1	5	1
0	2	0	1	8
0	4	0	0	4
1	0	3	0	0

Задание 9

$M =$

0	9	1	0	10
3	0	1	0	2
0	0	0	0	10
0	5	0	0	5
7	1	0	0	0

Задание 10

$M =$

0	0	9	10	11
1	0	2	12	13
0	0	0	1	1
0	0	0	0	7
5	0	0	3	0

Задание 11

$M =$

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
0	2	0	0	1
0	1	3	0	0

Задание 12

$M =$

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

Задание 13

$M =$

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	0

Задание 14

$M =$

0	0	1	9	0
1	0	1	2	10
0	0	0	1	17
0	1	0	0	1
1	0	0	4	0

Задание 15

$M =$

0	1	0	8	0
0	0	1	1	12
0	0	0	0	9
0	5	1	0	1
0	0	1	0	0

Задание 16

$M =$

0	2	1	0	0
1	0	0	1	11
0	1	0	1	1
1	1	0	0	9
1	0	0	3	0

Задание 17

$M =$

0	1	7	8	12
0	0	1	4	9
0	0	0	1	3
0	5	0	0	4
1	0	1	0	0

Задание 18

$M =$

0	3	4	7	15
1	0	0	1	8
2	1	0	1	1
0	0	0	0	4
3	0	0	1	0

Задание 19

$M =$

0	12	5	6	0
0	0	1	1	3
4	0	0	2	7
0	0	0	0	0
9	1	0	0	0

Задание 20

$M =$

0	0	15	2	11
1	0	4	1	7
0	9	0	1	8
0	0	0	0	0
1	5	3	2	0

Задание 21

$M =$

0	1	5	4	0
0	0	1	1	0
0	3	0	1	8
0	1	0	0	0
0	1	0	2	0

Задание 22

$M =$

0	0	2	1	7
1	0	1	4	3
0	0	0	1	4
0	0	2	0	5
1	0	6	0	0

Задание 23

$M =$

0	1	12	5	20
2	0	9	1	12
0	0	0	10	5
0	3	0	0	0
11	0	4	0	0

Задание 24

$M =$

0	0	4	7	5
10	0	11	1	3
3	0	0	1	12
0	0	0	0	6
2	0	8	4	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 7

Цель: практическое усвоение метода поиска минимального растущего дерева на взвешенном орграфе без контуров.

Длительность занятия – 2 часа.

Задача о поиске минимального растущего дерева с корнем в заданной вершине на взвешенном орграфе без контуров

Содержательная постановка задачи: требуется на заданном взвешенном ориентированном графе $G(X, U)$ выделить подграф – дерево $G'(X', U')$ с корнем в заданной s-ой вершине такой, что:

- Все вершины, достижимые из s-й вершины на $G(X, U)$, также достижимы из той же вершины на $G'(X', U')$.
- Суммарный вес дуг множества U' минимален.

Формальная постановка задачи:

$$\begin{cases} \sum\limits_{(i,j)\in U'} r(i,j)z(i,j) \to \min; \\ \forall x_t \in X : signum\left\{ \sum\limits_{d} \prod\limits_{L_d'(s,t)} z(i,j) \right\} = signum\left\{ \sum\limits_{d} \prod\limits_{L_d(s,t)} z(i,j) \right\}; \\ \prod\limits_{(i,j)\in U'} z(i,j) = 1; \\ \forall (i,j) \in U : z(i,j) = 1,0. \end{cases}$$

Свойства минимальных растущих деревьев:

- Величина $y = \sum\limits_{x_j \in X} \min\limits_{i} r(i,j)$ является нижней границей суммарного веса дуг минимального дерева.
- Если граф $G(X, U)$ не содержит контуров, то $y = \sum\limits_{x_j \in X'} \min\limits_{i} r(i,j)$ отвечает оптимальному значению целевой функции.

Алгоритм поиска минимального дерева на графе без контуров

Шаг 1. На исходном графе $G(X, U)$ удаляются все вершины, в которые отсутствуют пути из s-й вершины, являющейся корнем дерева. Полученный граф вновь обозначаем $G(X, U)$.

Шаг 2. $\forall j \neq s : r(p, j) = \min r(i, j)$.

Шаг 3. Дуги (p, j), определенные на предыдущем шаге, принадлежат множеству U'.

Шаг 4. Конец алгоритма.

Задания. Выделить на графе, заданном матрицей M, корневую вершину и найти минимальное растущее дерево.

Задание 1

$M =$

0	1	11	12	0
0	0	1	6	9
0	0	0	7	1
0	0	0	0	14
0	0	0	0	0

Задание 2

$M =$

0	0	10	11	15
3	0	1	17	7
0	0	0	1	6
0	0	0	0	5
0	0	0	0	0

Задание 3

$M =$

0	0	9	11	0
0	0	1	0	0
0	0	0	5	0
0	6	0	0	0
8	0	0	0	0

Задание 4

$M =$

0	18	19	0	15
0	0	17	0	16
0	0	0	0	13
14	0	0	0	12
0	11	0	0	0

Задание 5

$M =$

0	1	1	1	0
0	0	1	1	1
0	0	0	1	1
0	0	0	0	1
0	0	0	0	0

Задание 6

$M =$

0	0	13	12	11
10	0	16	15	14
0	0	0	18	17
0	0	0	0	19
0	0	0	0	0

Задание 7

$M =$

0	1	1	1	0
0	0	1	0	1
0	0	1	1	1
0	0	0	0	1
0	0	0	0	0

Задание 8

$M =$

0	0	10	19	0
13	0	14	15	16
0	0	0	17	18
0	0	0	0	12
11	0	0	0	0

Задание 9

$M =$

0	15	17	19	0
0	0	14	16	0
0	0	0	0	13
0	0	0	0	12
0	11	0	0	0

Задание 10

$M =$

0	0	11	12	13
14	0	0	15	16
0	17	0	18	19
0	0	0	0	20
0	0	0	0	0

Задание 11

$M =$

0	12	14	16	0
0	0	0	11	0
0	13	0	17	19
0	0	0	0	10
0	18	0	0	0

Задание 12

$M =$

0	0	5	10	7
11	0	1	0	2
0	0	0	1	4
0	10	0	0	3
0	0	0	0	0

Задание 13

$M =$

0	1	1	1	0
0	0	1	1	1
0	0	0	1	0
0	0	0	0	1
0	0	1	0	0

Задание 14

$M =$

0	0	1	1	0
1	0	1	0	1
0	0	0	1	1
0	1	0	0	1
0	0	0	0	0

Задание 15

$M =$

0	1	0	1	1
0	0	1	1	1
1	0	0	0	1
0	0	1	0	1
0	0	0	0	0

Задание 16

$M =$

0	0	1	0	1
1	0	0	1	1
0	1	0	1	1
1	0	0	0	1
0	0	0	0	0

Задание 17

$M =$

0	0	1	1	0
0	0	1	1	1
0	0	0	1	0
0	0	0	0	1
1	0	1	0	0

Задание 18

$M =$

0	0	1	1	1
0	0	0	1	1
0	1	0	1	1
0	0	0	0	0
0	0	0	1	0

Задание 19

$M =$

0	1	1	1	0
0	0	1	1	0
0	0	0	0	1
0	0	1	0	1
0	1	0	0	0

Задание 20

$M =$

0	0	1	0	1
1	0	1	1	1
0	0	0	1	1
1	0	0	0	1
0	0	0	0	0

Задание 21

$M =$

0	11	12	13	0
0	0	15	16	0
0	0	0	14	17
0	0	0	0	19
0	10	0	0	0

Задание 22

$M =$

0	0	11	10	0
15	0	19	0	13
0	0	0	16	17
0	0	0	0	18
12	0	0	0	0

Задание 23

$M =$

0	19	18	15	0
0	0	16	14	17
0	0	0	13	12
0	0	0	0	0
11	0	0	0	0

Задание 24

$M =$

0	0	10	19	18
11	0	17	16	15
0	0	0	13	14
0	0	0	0	12
0	0	0	0	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 8

Цель: практическое усвоение метода поиска минимального разреза на взвешенном бисвязном орграфе.

Длительность занятия – 2 часа.

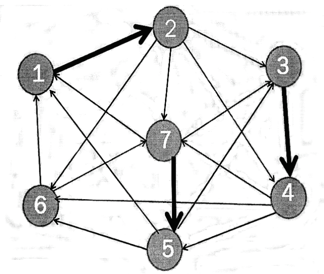
Задача о поиске минимального разреза на взвешенном сильносвязном орграфе

Содержательная постановка задачи: на взвешенном бисвязном ориентированном графе требуется выделить такое подмножество дуг, для которого справедливо:

1. Удаление дуг выделенного подмножества разрывает на графе все контуры.
2. Суммарный вес дуг выделенного подмножества минимален.

Пример 8.1.

Графовая интерпретация задачи о минимальном разрезе в бисвязном графе



Жирным выделены дуги одного из разрезов.

Формальная постановка задачи:

$$\begin{cases} \sum\limits_{i}\sum\limits_{j} r(i,j)z(i,j) \to \min; \\ \forall a_k \in A(G): \quad \sum\limits_{(i,j)\in U(a_k)} z(i,j) \ge 1; \\ \forall (i,j) \in U : z(i,j) = 1,0. \end{cases} \tag{8.1}$$

где $G(X, U)$ – взвешенный орграф;

X – множество вершин;

U – множество дуг;

$A(G)$ – множество контуров графа;

a_k – k-й контур множества $A(G)$

$X(a_k)$ – множество вершин k-го контура;

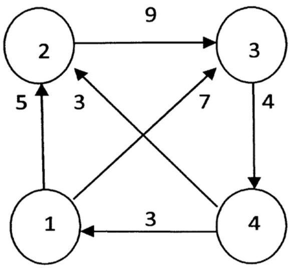
$U(a_k)$ – множество дуг k-го контура;

$z(i, j)$ – булева переменная;

$r(i, j)$ – вес дуги $(i, j) \in U$.

Тест графа на наличие контуров. Если последовательное удаление вершин-источников и вершин-стоков исчерпывает граф, то он был свободен от контуров.

Пример 8.2. Решить задачу о минимальном разрезе на графе вида:



Формальная постановка задачи примера 8.2:

$$\begin{cases} 5z(1,2) + 9z(2,3) + 4z(3,4) + 3z(4,1) + 3z(1,3) + 7z()4,2 \to \min; \\ z(1,2) + z(2,3) + z(3,4) + z(4,1) \geq 1; \\ z(1,3) + z(3.4) + z(4,1) \geq 1 \\ z(4,2) + z(2,3) + z(3,4) \geq 1; \\ \forall (i, j) \in U : z(i, j) = 1,0 \end{cases} \tag{8.2}$$

Решение задачи примера 8.2 перебором.

№	Z(4,2)	Z(1,3)	Z(4,1)	Z(3,4)	Z(2,3)	Z(1,2)	R
1	0	0	0	0	0	1	∞
2	0	0	0	0	0	0	∞
3	0	0	0	0	1	1	∞
4	0	0	0	0	1	0	∞
5	0	0	0	1	0	1	9
6	**0**	**0**	**0**	**1**	**0**	**0**	**4**
7	0	0	0	1	1	1	18
8	0	0	0	1	1	0	13
9	0	0	1	0	0	1	∞
10	0	0	1	0	0	0	∞
11	0	0	1	0	1	1	17

Ответ: $z(3,4)=1$;

$\forall (i, j) \in U \setminus (3,4) : z(i, j) = 0;$

$R = 4.$

Подмножество дуг $W \subset U$ является разрезом, если после удаления этих дуг последовательное отбрасывание вершин-источников и вершин-стоков исчерпывает граф.

$$5z(1,2) + 9z(2,3) + 4z(3,4) + 3z(4,1) + 3z(1,3) + 7z(4,2) \to \min;$$
$$z(1,2) + z(2,3) + z(3,4) + z(4,1) \geq 1;$$
$$z(1,3) + z(3,4) + z(4,1) \geq 1;$$
$$z(4,2) + z(2,3) + z(3,4) \geq 1;$$
$$\forall (i, j) \in U : z(i, j) = 1,0.$$

Достоинства и недостатки перебора:

Достоинства:

1. Гарантия глобально оптимального решения.
2. Простота алгоритма.
3. Легкость программной реализации.
4. Низкие требования к объему памяти компьютера
5. Легкость распараллеливания алгоритма.

Недостатки:

В ходе работы алгоритма генерируется более $2(2^n - 1)$ сочетаний различных чисел: алгоритм избыточен.

Задания. Пользуясь (5 x 5) матрицами M, как матрицами инциденций для ориентированных и взвешенных графов, определить на последних минимальные разрезы.

Задание 1

$M =$

0	1	1	1	10
2	0	1	1	12
0	7	0	1	4
4	0	3	0	1
0	0	2	5	0

Задание 2

$M =$

0	0	1	1	5
1	0	1	1	11
0	4	0	1	8
0	0	3	0	1
2	0	0	0	0

Задание 3

$M =$

0	0	1	1	4
0	0	1	0	5
7	0	0	1	6
0	1	5	0	3
1	0	0	3	0

Задание 4

$M =$

0	1	1	2	17
8	0	1	1	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

$M =$

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

Задание 6

$M =$

0	2	1	1	14
1	0	1	1	4
0	0	0	6	8
0	2	0	0	5
3	0	0	1	0

Задание 7

$M =$

0	7	1	2	10
2	0	1	0	1
0	0	0	1	6
1	4	0	0	1
5	0	0	3	0

Задание 8

$M =$

0	0	9	5	13
1	0	1	5	1
0	2	0	1	8
0	4	0	0	4
1	0	3	0	0

Задание 9

$M =$

0	9	1	5	10
3	0	1	1	2
0	0	0	0	10
0	5	0	0	5
7	1	0	0	0

Задание 10

$M =$

0	0	9	10	11
1	0	2	12	13
0	1	0	1	1
0	2	0	0	7
5	0	0	3	0

Задание 11

$M =$

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
4	2	0	0	1
0	1	3	0	0

Задание 12

$M =$

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

Задание 13

$M =$

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	0

Задание 14

$M =$

0	0	1	9	0
1	0	1	2	10
0	0	0	1	17
0	1	0	0	1
1	0	0	4	0

Задание 15

$M =$

0	1	0	8	0
0	0	1	1	12
1	0	0	0	9
0	5	1	0	1
1	0	1	0	0

Задание 16

$M =$

0	2	1	0	0
1	0	0	1	11
0	1	0	1	1
1	1	0	0	9
1	0	0	3	0

Задание 17

$M =$

0	1	7	8	12
0	0	1	4	9
0	0	0	1	3
0	5	0	0	4
1	0	1	0	0

Задание 18

$M =$

0	3	4	7	15
1	0	0	1	8
2	1	0	1	1
0	0	0	0	4
3	0	0	1	0

Задание 19

$M =$

0	12	5	6	0
0	0	1	1	3
4	0	0	2	7
0	0	1	0	9
9	1	0	0	0

Задание 20

$M =$

0	0	15	2	11
1	0	4	1	7
0	9	0	1	8
1	0	0	0	9
1	5	3	2	0

Задание 21

$M =$

0	1	5	4	0
4	0	1	1	0
0	3	0	1	8
0	1	0	0	6
0	1	0	2	0

Задание 22

$M =$

0	0	2	1	7
1	0	1	4	3
0	0	0	1	4
0	1	2	0	5
1	0	6	0	0

Задание 23

$M =$

0	1	12	5	20
2	0	9	1	12
0	0	0	10	5
0	3	0	0	6
11	0	4	0	0

Задание 24

$M =$

0	3	4	7	5
10	0	11	1	3
3	0	0	1	12
0	7	0	0	6
2	0	8	4	0

ПРАКТИЧЕСКОЕ ЗАНЯТИЕ № 9

Цель: практическое усвоение метода поиска решения задачи коммивояжера на взвешенном бисвязном орграфе.

Длительность занятия – 2 часа.

Задача о поиске решения задачи коммивояжера на взвешенном сильносвязном орграфе

Содержательная постановка задач:

1. **Разомкнутая постановка задачи**: коммивояжер должен объехать все n городов, побывав в каждом по одному разу, и затратив минимум средств на путешествие.

2. **Замкнутая постановка задачи**: коммивояжер должен объехать все n городов, побывав в каждом по одному разу и вернуться в город из которого стартовал, затратив минимум средств на путешествие.

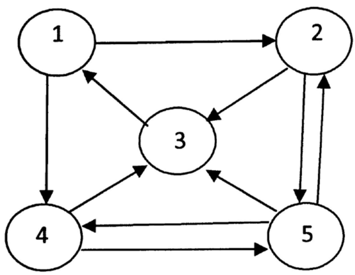


Рис. 9.1. Граф $G(X, U)$.

Гамильтоновы контуры на G(X,U):

$a_1 = 1, 2, 5, 4, 3, 1;$

$a_2 = 5, 2, 3, 1, 4, 5.$

Разомкнутые решения на G(X,U):

$L_1 = 1, 2, 5, 4, 3;$

$L_2 = 5, 2, 3, 1, 4.$

Формальная постановка замкнутой задачи:

$$\begin{cases} \sum\limits_i \sum\limits_j r(i,j) z(i,j) \to \min; \\ \forall a_k \in A(G) : \left| X(a_k) \right| < \left| X \right| \Rightarrow \prod\limits_{(i,j) \in U(a_k)} z(i,j) = 0; \\ \forall x_j \in X : \sum\limits_i z(i,j) = 1; \\ \forall x_q \in X : \sum\limits_i z(q,i) = 1; \\ \forall (i,j) \in U : z(i,j) = 1,0, \end{cases}$$

где $G(X, U)$ – взвешенный орграф;

X – множество вершин;

U – множество дуг;

$A(G)$ – множество контуров графа;

a_k – k-й контур множества $A(G)$

$X(a_k)$ – множество вершин k-го контура;

$U(a_k)$ – множество дуг k-го контура;

$z(i, j)$ – булева переменная;

Формальная постановка разомкнутой задачи коммивояжера:

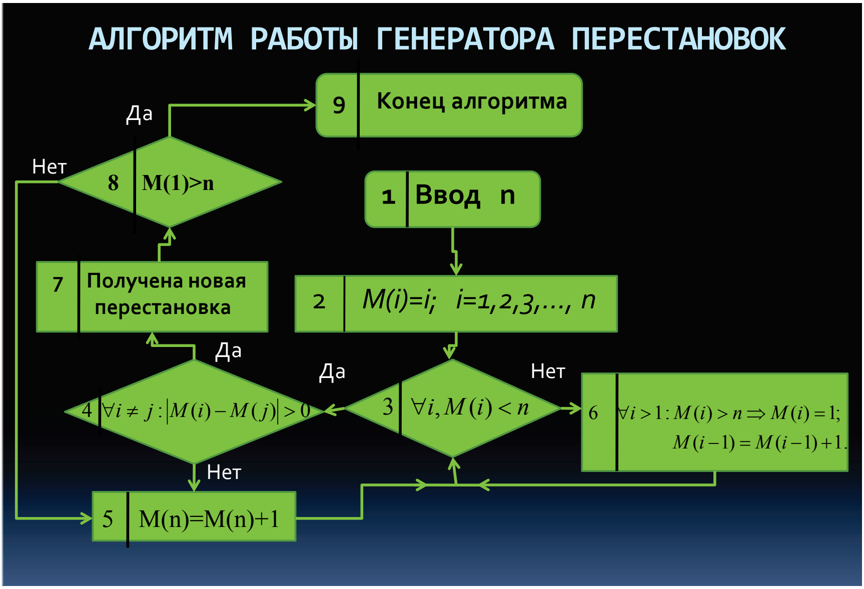
$$\begin{cases} \sum_i \sum_j r(i,j) z(i,j) \to \min; \\ \forall a_k \in A(G): \quad \prod_{(i,j)\in U(a_k)} z(i,j) = 0; \\ \forall x_j \in X \setminus (x_s \bigcup x_t): \sum_i z(i,j) = \sum_k z(j,k) = 1; \\ \sum_i z(s,i) = 1; \\ \sum_i z(i,t) = 1; \\ \sum_i z(i,s) = \sum_k z(t,k) = 0; \\ \forall (i,j) \in U: z(i,j) = 1,0. \end{cases}$$

Решение задачи коммивояжера с помощью генератора перестановок

Принцип работы генератора перестановок:

№	1-е число	2-е число	3-е число	Перестановка
1	1	2	3	+
2	1	3	1	–
3	1	3	2	+
4	1	3	3	–
5	2	1	1	–

6	2	1	2	–
7	**2**	**1**	**3**	+
8	2	2	1	–
9	2	2	2	–
10	2	2	3	–
11	2	3	1	+



Достоинства и недостатки решения задачи коммивояжера с помощью генератора перестановок.

Достоинства:

1. Полученное решение является глобально оптимальным.
2. Простота алгоритма.
3. Легкость программной реализации.
4. Низкие требования к объему памяти компьютера

Недостатки:

1. В ходе работы алгоритма генерируется более n! сочетаний различных чисел: алгоритм избыточен.
2. Сложность распараллеливания алгоритма.

Задания. Пользуясь (5 х 5) матрицами M, как матрицами инциденций для ориентированных, взвешенных и сильносвязных графов, определить на последних решения замкнутых и разомкнутых задач коммивояжера.

Задание 1

$M =$

0	1	1	1	10
2	0	1	1	12
0	7	0	1	4
4	0	3	0	1
0	0	2	5	0

Задание 2

$M =$

0	0	1	1	5
1	0	1	1	11
0	4	0	1	8
0	0	3	0	1
2	0	0	0	0

Задание 3

$M =$

0	0	1	1	4
0	0	1	0	5
7	0	0	1	6
0	1	5	0	3
1	0	0	3	0

Задание 4

$M =$

0	1	1	2	17
8	0	1	1	3
0	4	0	0	1
1	0	2	0	1
3	7	0	0	0

Задание 5

$M =$

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

Задание 6

$M =$

0	2	1	1	14
1	0	1	1	4
0	0	0	6	8
0	2	0	0	5
3	0	0	1	0

Задание 7

$M =$

0	7	1	2	10
2	0	1	0	1
0	0	0	1	6
1	4	0	0	1
5	0	0	3	0

Задание 8

$M =$

0	0	9	5	13
1	0	1	5	1
0	2	0	1	8
0	4	0	0	4
1	0	3	0	0

Задание 9

$M =$

0	9	1	5	10
3	0	1	1	2
0	0	0	0	10
0	5	0	0	5
7	1	0	0	0

Задание 10

$M =$

0	0	9	10	11
1	0	2	12	13
0	1	0	1	1
0	2	0	0	7
5	0	0	3	0

Задание 11

M =

0	1	1	1	0
7	0	0	1	0
0	1	0	1	1
4	2	0	0	1
0	1	3	0	0

Задание 12

M =

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

Задание 13

M =

0	1	1	1	0
5	0	1	1	1
0	2	9	1	0
3	0	0	0	1
2	0	1	1	0

Задание 14

M =

0	0	1	9	0
1	0	1	2	10
0	0	0	1	17
0	1	0	0	1
1	0	0	4	0

Задание 15

M =

0	1	0	8	0
0	0	1	1	12
1	0	0	0	9
0	5	1	0	1
1	0	1	0	0

Задание 16

M =

0	2	1	0	0
1	0	0	1	11
0	1	0	1	1
1	1	0	0	9
1	0	0	3	0

Задание 17

M =

0	1	7	8	12
0	0	1	4	9
0	0	0	1	3
0	5	0	0	4
1	0	1	0	0

Задание 18

M =

0	3	4	7	15
1	0	0	1	8
2	1	0	1	1
0	0	0	0	4
3	0	0	1	0

Задание 19

M =

0	12	5	6	0
0	0	1	1	3
4	0	0	2	7
0	0	1	0	9
9	1	0	0	0

Задание 20

M =

0	0	15	2	11
1	0	4	1	7
0	9	0	1	8
1	0	0	0	9
1	5	3	2	0

Задание 21

M =

0	1	5	4	0
4	0	1	1	0
0	3	0	1	8
0	1	0	0	6
0	1	0	2	0

Задание 22

M =

0	0	2	1	7
1	0	1	4	3
0	0	0	1	4
0	1	2	0	5
1	0	6	0	0

Задание 23

M =

0	1	12	5	20
2	0	9	1	12
0	0	0	10	5
0	3	0	0	6
11	0	4	0	0

Задание 24

M =

0	3	4	7	5
10	0	11	1	3
3	0	0	1	12
0	7	0	0	6
2	0	8	4	0

Литература

1. *Гроппен В. О.* Экстремальные задачи на взвешенных графах .Изд. Фламинго, Владикавказ, 2012, 92 с.
2. *Бурков В. Н., Новиков Д. А.* Элементы теории графов. 2008, http://dmtsoft.ru/bn/391/as/oneaticleshablon/ .
3. *Костюкова Н. И.* Графы и их применение. Комбинаторные алгоритмы для программистов. Учебное пособие. М.: Интернет-Университет Информационных технологий: Бином, Лаборатория знаний, 2007.