

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
НОВГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ ЯРОСЛАВА МУДРОГО**

**РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ НА
ПЭВМ**

**ВЕЛИКИЙ НОВГОРОД
2017**

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ
НОВГОРОДСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
ИМЕНИ ЯРОСЛАВА МУДРОГО

РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ НА ПЭВМ

Методические указания к практическим занятиям

Великий Новгород
2017

УДК 656.13.07.:681.31

Печатается по решению
РИС НовГУ

Рецензенты:

кандидат технических наук, доцент А.Н. Чадин
кандидат технических наук, доцент И. И. Зубрицкас

Решение транспортной задачи на ПЭВМ: Метод. указания к практическим занятиям / Сост. С. В. Гудилов; НовГУ им. Ярослава Мудрого. – Великий Новгород; 2017. – 50с.

Изложена постановка транспортной задачи и способ её решения с помощью метода линейного программирования. Приводятся алгоритм и пример программы нахождения оптимального решения транспортной задачи с помощью ПЭВМ.

Методические указания предназначены для студентов специальности 23.03.03 "Эксплуатация транспортно-технологических машин и комплексов" к модулю «Основы инженерных расчетов».

УДК 656.13.07.:681.31

© Новгородский государственный
университет, 2017

© С. В. Гудилов составление, 2017

ОГЛАВЛЕНИЕ

ВВЕДЕНИЕ.....	5
1. ЦЕЛЬ РАБОТЫ	6
2. РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ МЕТОДОМ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ	6
2.1. Постановка задачи и её решение	6
2.2. Алгоритм последовательного улучшения плана.....	12
2.3. Дисбаланс и вырожденность в транспортной задаче	15
2.4. Постановка транспортной задачи на ПЭВМ	17
3. ПОСЛЕДОВАТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ РАБОТЫ	22
4. СОДЕРЖАНИЕ ОТЧЁТА.....	22
5. ТЕХНИКА БЕЗОПАСНОСТИ.....	22
6. ПРИМЕРЫ ЗАДАНИЙ.....	22
БИБЛИОГРАФИЧЕСКИЙ СПИСОК	25
Приложение 1	26
Приложение 2	326

ВВЕДЕНИЕ

Снижение транспортных издержек – большая общегосударственная задача. Путь к решению – рациональное размещение производственных сил: разработка оптимальных схем грузопотоков, исключение встречных перевозок. Реализация этих задач требует дальнейшего совершенствования планирования на автомобильном транспорте. Значительное внимание при этом должно быть уделено применению в планировании экономико-математических методов. Целью применения этих методов является выбор из многих возможных вариантов плана оптимального, т.е. наилучшего с точки зрения эффективности.

В настоящее время при оперативном планировании перевозок грузов и пассажиров, при анализе деятельности автотранспортных предприятий и объединений используются методы математического программирования, теории массового обслуживания, имитационного моделирования, математико-статистические методы и другие.

Поставленные задачи в современных условиях, когда требуется оперативно обработать большое количество данных и выбрать наилучший вариант, невозможно решить без применения ПЭВМ. Применение ПЭВМ на автотранспортных предприятиях позволяет вывести планирование на совершенно новый уровень, поднять производительность труда плановых отделов и отрасли в целом, сделать труд в плановых отделах более квалифицированным и современным.

1. ЦЕЛЬ РАБОТЫ

Основная цель работы: научиться находить оптимальное решение транспортной задачи с помощью ПЭВМ.

2. РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ МЕТОДОМ ЛИНЕЙНОГО ПРОГРАММИРОВАНИЯ

2.1. Постановка задачи и её решение

Управление строительством должно отправить некоторое количество железобетонных блоков на пять объектов с трёх складов. На складах имеется соответственно 15, 25, 20 блоков, а на объектах требуется соответственно 20, 12, 5, 8 и 15. Стоимость перевозки одного блока (в рублях) со склада на объект приведена в табл. 2.1.

Таблица 2.1.

Стоимости перевозок

Склад	Объект				
	S ₁	S ₂	S ₃	S ₄	S ₅
W ₁	1	0	3	4	2
W ₂	5	1	2	3	3
W ₃	4	8	1	4	3

Следует спланировать перевозку со складов на объекты железобетонных блоков так, чтобы стоимость перевозки была минимальная.

Пусть x_{ij} – количество блоков, отправляемых со склада i на объект j . Ясно, что $x_{ij} \geq 0$, и в силу ограничений на возможности поставки со складов (предложение) и спрос на объектах они удовлетворяют следующим условиям:

для предложения:

$$\begin{aligned}x_{11} + x_{12} + x_{13} + x_{14} + x_{15} &= 15 \\x_{21} + x_{22} + x_{23} + x_{24} + x_{25} &= 25 \\x_{31} + x_{32} + x_{33} + x_{34} + x_{35} &= 20;\end{aligned}$$

для спроса:

$$\begin{aligned}x_{11} + x_{21} + x_{31} &= 20 \\x_{12} + x_{22} + x_{32} &= 12 \\x_{13} + x_{23} + x_{33} &= 5 \\x_{14} + x_{24} + x_{34} &= 8 \\x_{15} + x_{25} + x_{35} &= 15.\end{aligned}$$

Стоимость, равная

$$C = x_{11} + 0x_{12} + 3x_{13} + 4x_{14} + 2x_{15} + \dots + 4x_{34} + 3x_{35}$$

должна быть минимизирована при этих ограничениях.

Эта задача является задачей линейного программирования, и поставленные условия обобщаются на транспортную задачу с m пунктами производства и объёмами производства a_i ($i = 1, 2, 3, \dots, m$), и n пунктами потребления и объёмами потребления b_j ($j = 1, 2, 3, \dots, n$), где:

$$\sum_{i=1}^m a_i = \sum_{j=1}^n b_j. \quad (2.1)$$

Если C_{ij} – стоимость транспортировки одного изделия из пункта производства i в пункт потребления j , то задача заключается в нахождении $x_{ij} \geq 0$, удовлетворяющим следующим соотношениям:

$$\begin{aligned} x_{11} + x_{12} + \dots + x_{1n} &= a_1, \\ x_{21} + \dots + x_{2n} &= a_2, \\ &\dots \\ x_{m1} + \dots + x_{mn} &= a_m, \end{aligned} \quad (2.2)$$

$$\begin{aligned} x_{11} + x_{21} + \dots + x_{m1} &= b_1, \\ x_{12} + x_{22} + \dots + x_{m2} &= b_2, \\ &\dots \\ x_{1n} + x_{2n} + \dots + x_{mn} &= b_n \end{aligned}$$

и минимизирующей функцию:

$$C = c_{11}x_{11} + c_{12}x_{12} + \dots + c_{mn}x_{mn}$$

Соотношения (2.2) можно переписать так:

найти такие $x_{ij} \geq 0$, для которых справедливы неравенства:

$$\sum_{j=1}^n x_{ij} = a_i > 0, (i = \overline{1, m}) \quad (2.3)$$

$$\sum_{i=1}^m x_{ij} = b_j > 0, (j = \overline{1, n}) \quad (2.4)$$

и некоторые минимизируют функцию

$$C = \sum_{i=1}^m \sum_{j=1}^n c_{ij} x_{ij} \quad (2.5)$$

Поскольку

$$\sum_{i=1}^m a_i = \sum_{i=1}^m \sum_{j=1}^n x_{ij} = \sum_{j=1}^n \sum_{i=1}^m x_{ij} = \sum_{j=1}^n b_j$$

согласно уравнению (2.1) имеется всего $m+n-1$ независимых ограничений и, следовательно, $m+n-1$ базисных переменных в базисном допустимом решении.

Удобнее не рассматривать ограничения, а работать с массивом транспортных данных в виде, приведённом ниже. Следует разместить неотрицательные переменные в клетках таким образом, чтобы сумма по строкам и столбцам совпадали с указанными правыми частями ограничений в виде равенств примера и чтобы сумма произведений этих переменных на стоимости (указанные в правом нижнем углу каждой клетки) была минимальна. Приведённый на рис. 2.1 массив соответствует данным примера:

1	0	3	4	2	15
5	1	2	3	3	25
4	2	1	4	3	20
20	12	5	8	15	

Рис. 2.1

Представляя данные в таком виде, легко построить первое базисное допустимое решение задачи. Это можно сделать по правилу "самая дешёвая продукция реализуется первой". Поскольку задача состоит в минимизации общей стоимости, находим наименьшую стоимость во всех клетках: 0 в строке 1 столбца 2 и присваиваем переменной x_{12} значение 12 (наименьшая из сумм по строке и по столбцу). Теперь столбец 2 можно удалить, уменьшив сумму по строке на 12, т.е. заменив её на 3. Далее та же процедура применяется к полученному массиву (рис. 2.2).

1	12	0	3	4	2	15 3
5		1	2	3	3	25
4		2	1	4	3	20
20	12		5	8	15	

Рис. 2.2

Затем переменной x_{11} присваивается значение 3 (или переменной x_{33} – значение 5), удаляется строка 1, сумма по столбцу 1 заменяется на 17 и осуществляется переход к следующему массиву и так далее (рис. 2.3).

3	12	3				
1	0		4	2	3	
5	1	2	3	3	25	
4	2	1	4	3	20	
20	17	12	5	8	15	

Рис. 2.3

После небольшой тренировки для задач небольшого объёма эту процедуру можно проводить устно. После того, как последней переменной присвоено определённое значение, суммы по всем строкам и столбцам будут равны 0. Таким образом, получается решение (рис. 2.4, значения переменных находятся в левых верхних углах клеток) с семью, приводимыми ниже, базисными переменными. Остальные переменные равны 0. Для общего массива на m строк и n столбцов получаем $m+n-1$ переменных в силу (2.1).

3	12					
1	0	3	4	2	15	
2			8	15	3	25
5	1	2	3	3		
15		5				20
4	8	1	4	3		
20	12	5	8	15		

Рис. 2.4

Полная стоимость, соответствующая этому решению,

$$C=3*1+12*0+2*5+8*3+15*3+15*4+5*1=147 \text{ рублей.}$$

Попробуем теперь улучшить это решение, уменьшив стоимость C . Отметим, что полученные результаты для этого частного случая и метод их получения применимы и в случае общей транспортной задачи.

Пусть c_{sr} имеет наименьшее значение из всех c_{ij} . Положим x_{sr} равным наименьшим из величин a_s и b_r . Если этой величиной будет a_s , то удалим строку S , заменим b_r на $b_r - a_s$ и применим вышеописанную процедуру к оставшемуся массиву. Полученное таким образом базисное решение будет содержать $m+n-1$ базисных переменных, и каждая из этих переменных x_{ij} будет задаваться соотношением:

$$x_{ij} = \pm \sum a_p \pm \sum b_q .$$

Поскольку базис задаётся треугольной системой уравнений, по крайней мере, одна строка или один столбец содержат единственную переменную.

Поэтому $x_{pq}=a_p$ (строка p) или $x_{pq}=b_q$ (столбец q)

Если выполняется первое из этих равенств, т.е. $a_p < b_q$, то удаляется строка p , а b_q заменяется на $b_q - a_p$. Затем рассуждения повторяются для оставшегося массива, т.е. полагается, что

$$x_{p'q'} = a_{p'} \text{ или } x_{p'q'} = b_{q'}, \text{ или } x_{p'q'} = b_{q'} - a_{p'}.$$

Повторив эту процедуру несколько раз, увидим, что все базисные переменные имеют вид

$$\sum a_p - \sum b_q \text{ или } -\sum a_p + \sum b_q$$

Заметим, что если все a_i или b_j — целые, значения базисных переменных в допустимом базисном решении тоже целые. Поскольку эта задача линейного программирования, оптимальное решение является базисным допустимым решением и, следовательно, целым. Это гарантирует, что в нашем примере не будет получено абсурдное решение (3/4 блока).

2.2. Алгоритм последовательного улучшения плана

Будем решать транспортную задачу с помощью алгоритма последовательного улучшения, разработанного Ф.Л. Хитикоком [2].

Для задачи базисное допустимое решение, полученное по правилу "самая дешёвая продукция реализуется первой", было построено (рис. 2.4).

Предположим, что для рассмотренной задачи уже построено допустимое базисное решение, в котором некоторые из переменных отличны от 0, а остальные переменные — небазисные и, следовательно, равны 0. Если $-u_i$ и v_j — симплекс-множители для ограничений, соответствующих i -й строке и j -му столбцу в этом базисе, то после умножения i -й строки на $-u_i$, j -го столбца на $-v_j$ и прибавления стоимости C получаем

$$\begin{aligned} x_{11} + x_{12} + \dots + x_{1n} &= a_1(x - u_1), \\ x_{21} + \dots + x_{2n} &= a_2(x - u_2), \\ &\dots \\ x_{m1} + \dots + x_{mn} &= a_m(x - u_m), \\ x_{11} + x_{21} + \dots + x_{m1} &= b_1(x - v_1), \\ x_{12} + x_{22} + \dots + x_{m2} &= b_2(x - v_2), \\ &\dots \\ x_{1n} + x_{2n} + \dots + x_{mn} &= b_n(x - v_n), \\ c_{11}x_{11} + c_{12}x_{12} + \dots + c_{mn}x_{mn} &= C \end{aligned}$$

Следовательно,

$$\sum_{i=1}^m \sum_{j=1}^n (c_{ij} - u_i - v_j) = C - \sum_{i=1}^m u_i a_i - \sum_{j=1}^n v_j b_j \quad (2.7)$$

Коэффициент при x_{ij} равен $c_{ij} - u_i - v_j$, так как x_{ij} входит всего в два ограничения, соответствующих i -й строке и j -му столбцу.

Если уравнение (2.7) является канонической формой целевой функции, соответствующей базису, то коэффициенты при базисных переменных равны 0.

Таким образом, для занятых клеток массива справедливо соотношение

$$c_{ij} - u_i - v_j = 0 \quad (2.8)$$

Следовательно, можно определить u_i и v_j .

Имеется m неизвестных u_i и n неизвестных v_j и, поскольку базисными переменными занято $m+n-1$ клеток, из уравнения (2.8) получаем $m+n-1$ уравнений $m+n$ неизвестных. Эти уравнения можно решить, положив одно из неизвестных равным 0 и решив систему относительно остальных неизвестных. В нашем примере на первом шаге имеем следующее базисное допустимое решение (рис. 2.5).

3+w	12-w				15(-4)
1	0	3	4	2	
2-w	w	8	15		25(0)
5	-3 1	0 2	3	3	C=147
15		5			20(-1)
4	8	1	4	3	
20(5)	12(4)	5(2)	8(3)	15(3)	

Рис. 2.5

Имеется 8 неизвестных u_1, u_2, u_3 и v_1, v_2, \dots, v_5 и 7 занятых клеток. Если положить $u_2 = 0$, то в трёх занятых клетках этой строки получим $v_1 = 5, v_4 = 3$ и $v_5 = 3$. Поскольку $v_1 = 5$, то в занятых клетках этого столбца $u_1 = -4$ и $u_3 = -1$. Наконец, так как $u_1 = -4$ и $u_3 = -1$, то $v_2 = 4$ и $v_3 = 2$. Таким образом, получаются указанные в скобках на рис. 2.5 значения u_i и v_j .

Теперь можно проверить, оптимально ли это решение. Если c'_{ij} – коэффициент при x_{ij} в канонической форме для целевой функции, то из уравнений (2.7) следует, что

$$c'_{ij} = c_{ij} - u_i - v_j$$

Для базисных переменных $c_{ij} = 0$. Для небазисных переменных отрицательное значение c'_{ij} указывает на то, что переменная x_{ij} должна быть включена в базисные переменные, что приведёт к уменьшению целевой функции. В связи с этим c'_{ij} вычисляется для незанятых клеток; те из них, в которых значение c'_{ij} отрицательно, помечаются. Результат указан в левых нижних углах массива (рис. 2.5). На этом этапе нулевые значения переменных указывают на то, что их введение в базис не изменит значение целевой функции.

Ясно, что для нашей задачи увеличение x_{22} приведёт к уменьшению целевой функции. Действительно, каждое увеличение x_{22} на 1 уменьшит целевую функцию на 3. При увеличении значения x_{22} на число w необходимо уменьшить x_{21} на число w , чтобы сохранить сумму в строке (2). Для того чтобы сохранить сумму в столбце (1), необходимо увеличить x_{11} на число w , а затем для сохранения суммы в строке 1 уменьшить x_{12} на число w . Заметим, что другой метод (положить $x_{22} = w$, уменьшить x_{42} на число w до значения $8-w$, поскольку невозможно сохранить сумму в столбце (4), не вводя дополнительных переменных) – приводит к некоторым трудностям и с его помощью невозможно получить базисное решение.

Таким образом, максимальное значение, которое может иметь w , равно 2. В этом случае переменная x_{21} становится небазисной и нулевой в следующем случае допустимом базисном решении (рис. 2.6)

5+ w 1	10- w 0	3	4	0	2	15(-1)
5 5	2+ w 1	2	8	3	15- w 3	25(0)
15- w 4	8	5 1	-1	4	w -2	20(2)
20(2)	12(1)	5(-1)	8(3)	15(3)		

Рис. 2.6

Для этого массива (рис. 2.6):

$$C = 5*1 + 10*0 + 2*1 + 8*3 + 15*3 + 15*4 + 5*1 = 141 = 147 - 3*2 = \\ = (\text{предыдущее значение}) + c'_{22}*w$$

Далее определяем u_i и v_j для этого решения. Значения c'_{ij} для небазисных переменных, равных 0 или отрицательных, указаны в левых нижних углах клеток массива. Очевидно, что в базис следует включить переменную x_{35} . Если изменить эту величину на число w , то другие

величины должны быть заменены указанным способом. В результате наибольшее возможное значение для w равно 10, и это приводит к тому, что в следующем массиве переменная x_{12} станет небазисной (рис. 2.7)

15 1	0	3	4	2	15(-3)
5 5	12 1	2	8 3	5 3	25(0)
15 4	8	5 1	4	10 3	20(0)
20(4)	12(1)	5(-1)	8(3)	15(3)	

Рис. 2.7

Значение стоимости C уменьшается:

$$C = 15*1 + 12*1 + 8*3 + 5*3 + 5*4 + 5*1 + 10*3 = 121 = 141 - 2*10 = \\ = (\text{предыдущее значение}) + c'_{35}*w.$$

Для этого массива вычисляются u_i и v_j . Для незанятых клеток все c'_{ij} положительны. Таким образом, получено оптимальное решение, в котором $x_{11} = 15$, $x_{22} = 12$, $x_{24} = 8$, $x_{25} = 5$, $x_{31} = 5$, $x_{33} = 5$, $x_{35} = 10$.

Минимальное значение стоимости C равно 121 рубль.

Отметим, что значение стоимости C задаётся уравнением

$$C = \sum_{i=1}^m a_i u_i + \sum_{j=1}^n v_j b_j, \quad (2.10)$$

что непосредственно следует из уравнения (2.7), так как в его левой части все слагаемые равны 0.

2.3. Дисбаланс в транспортной задаче

Выполнение условия (2.1) очень важно в транспортной задаче. Для массива размерностью $m+n$ оно означает, что в базисное допустимое решение входит $m+n-1$ базисная переменная. Предположим, что баланс между спросом и предложением нарушен.

Пусть 15, 25, 20 железобетонных блоков, хранящихся на складах W_1 , W_2 , W_3 должны быть распределены на 4 объекта, где требуется 20, 12, 5, 9 блоков. Стоимость перевозки одного блока со склада на объект задана в табл. 2.2

Таблица 2.2

Склад	Объект			
	S_1	S_2	S_3	S_4
W_1	2	2	2	4
W_2	3	1	1	3
W_3	3	6	3	4

Склады располагают 60 блоками, а на объектах их требуется всего 46. Для того чтобы перейти к транспортной задаче, введём фиктивный объект, которому требуется 14 блоков. Стоимость перевозки со склада на этот объект 1 блока полагается равной 0 рублей. Если в окончательном решении будет получено, что некоторые блоки надо будет отправить на этот объект, то это будет проигнорировано. Блоки останутся на складе. Таким образом, поставлена транспортная задача, для которой уравнения (2.1) выполняются.

На рисунке 2.8 приводится первое базисное решение для этого массива и последующие массивы, ведущие к окончательному решению.

15						
2	2	2	4	0	15(-1)	
5-w	12	5	3+w		25(0)	$C=75$
3	1	1	3	0		
w			6-w	14	20(1)	
-1	3	6	3	4	0	
20(1)	12(1)	5(1)	9(3)	14(-1)		

Рис. 2.8

Переменная x_{31} входит в базис; максимальное значение $w = 5$; переменная x_{31} исключается из базиса:

15						
2	2	2	4	0	15(0)	
3	12	5	8		25(0)	$C=90$
	1	1	3	0		
5				14	20(1)	
3	6	3	4	0		
20(2)	12(1)	5(1)	9(3)	14(-1)		

Рис. 2.9

14 блоков из клетки (3.5) остаются на складе 3. Потребности объектов полностью удовлетворены. Мы получили оптимальное решение (рис. 2.9)

$$x_{11} = 15, x_{22} = 12, x_{23} = 5, x_{24} = 8, x_{31} = 5, x_{34} = 1;$$

$$C = 90 \text{ рублей.}$$

Таким образом, для того, чтобы справиться с дисбалансом (спрос превышает предложение) надо ввести фиктивного поставщика с нулевой стоимостью перевозок. Продукция этого поставщика на самом деле поставляться не будет.

2.4. Постановка транспортной задачи на ПЭВМ

Общая блок-схема решения транспортной задачи приведена на рис. 2.10.

Блок-схема определения первого допустимого базисного решения приведена на рис. 2.11.

В конце этой процедуры все элементы массива $D0(I)$ и $D1(J)$ (приложение 1) должны быть равны 0. Переменные $T0(I)$ и $T1(I)$ должны быть равны количеству переменных соответственно в I строке и J -м столбце.

В следующей процедуре вычисляются u и v и наименьшее значение c'_{ij} , предположим c'_{kl} (рис. 2.12).

Поясним процедуру перехода к новому базисному допустимому решению подробнее. Сначала находим "цепь" $\pm w$ клеток, которая не является "тупиковым путём".

На шаге I мы находимся в клетке (K, L) , T – счётчик шагов, 15 – индикатор "тупикового пути", $(R0(T), C3(T)) = (R1, C2)$ – клетка, в которую мы попадаем на шаге T . Массив D состоит из ± 1 , соответствующих $\pm w$; положим $M0 = 1$, если клетка используется, 12 = 1 и 13 = 1, если строки и столбцы входят в цикл.

В команде 2102 на шаге T ищется строка $R0(T)$ для столбца, содержащего базисную переменную в неиспользованном столбце (строка 2140) в неотмеченной клетке (строка 2150). Если это единственная переменная в своём столбце, то производится присвоение 15 = 1. Разумеется, это не делается в начальном столбце. После того, как подходящая переменная найдена в столбце $C2$, поиск прекращается; при этом $F1 = 1$.

Затем T увеличивается для следующего шага, в переменную $R0(T)$ заносится номер текущей строки, а в переменную $C3(T)$ – номер только что найденного столбца $C2$; соответствующему D присваивается значение -1, и найденная клетка помечается присвоением $M0$ значения 1. Если мы снова оказались в столбце L , откуда начали, то цикл завершён. В противном

случае ищем столбец $C3(T)[=C2]$ для строки, содержащей базисную переменную в неотмеченных строке и клетке; таким образом, снова помечаются "тупиковые пути". Как только искомым столбец найден, поиск прекращается присвоением $F0 = 1$. Затем T увеличивается для следующего шага, переменной $R0(T)$ присваивается номер только что найденной строки $R1$, а $C3(T)$ – номер только что обрабатывавшегося столбца, для этой клетки осуществляются присвоения: $D = +1$, $M0 = 1$, после чего программа возвращается к поиску строки в строке 2100 программы.

Заметим, что если в процессе поиска строки не удаётся найти столбец ($C2 = 0$ в строке 2190), не являющийся "тупиковым путём", то происходит возвращение (строка 2210) к строке предыдущего поиска столбца. Если в поисках столбца удаётся найти только строки ($R1 = 0$ в строке 2590), соответствующие "тупиковым путям", осуществляется возвращение (строка 2610) к строке предыдущего поиска строки. Однако в силу того, что $M0$ сохраняет своё значение, ошибка не повторяется в дальнейшем ($M0 = 1$ в строках 2150 и 2550). Поскольку базис задан треугольной системой уравнений, процесс, в конце концов, закончится, и управление будет передано из строки 2400 в строку 3000.

В строках 3000-3040 программы содержится наименьшая базисная из клеток, в которой $D = -1$. Здесь определяются значение и положение переменной $(K0, L0)$, которая будет удалена из базиса. В строках 3100-3120 клетки включаются в цепь. В конечном счёте, переменная (K, L) определяется как базисная, переменная $(K0, L0)$ – как небазисная и определяется количество базисных переменных во всех строках и столбцах. Затем программа возвращается к вычислению симплекс-множителей u и v .

Программа решения транспортной задачи на примере, описанном в разделе 2.1, приведена в приложениях 1 и 2. Результаты расчётов на ПЭВМ и расчётов вручную совпадают, хотя описанный алгоритм решения не соответствует пути решения задачи вручную из-за вырожденности первых двух из полученных допустимых базисных решений.

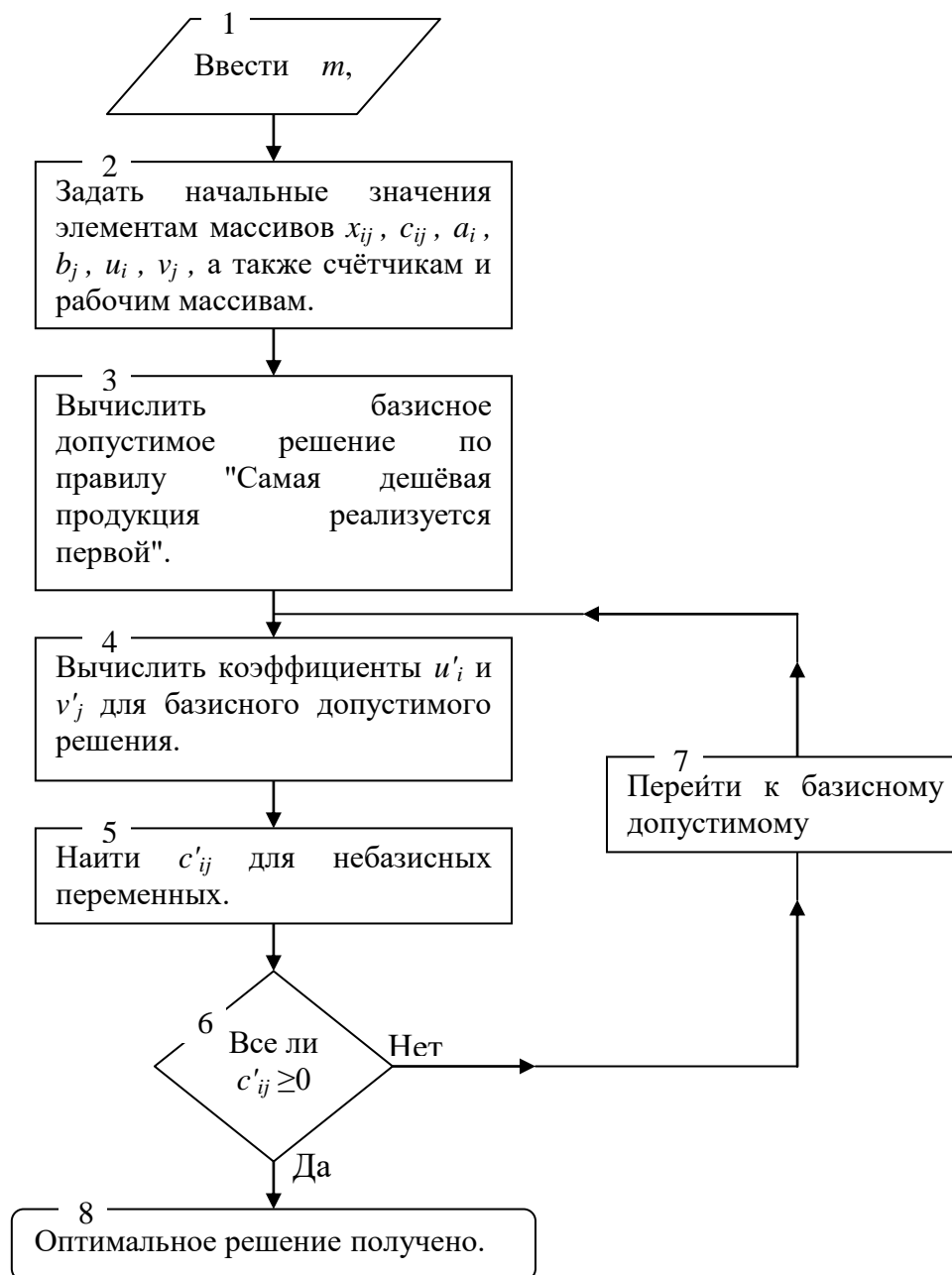


Рис. 2.10

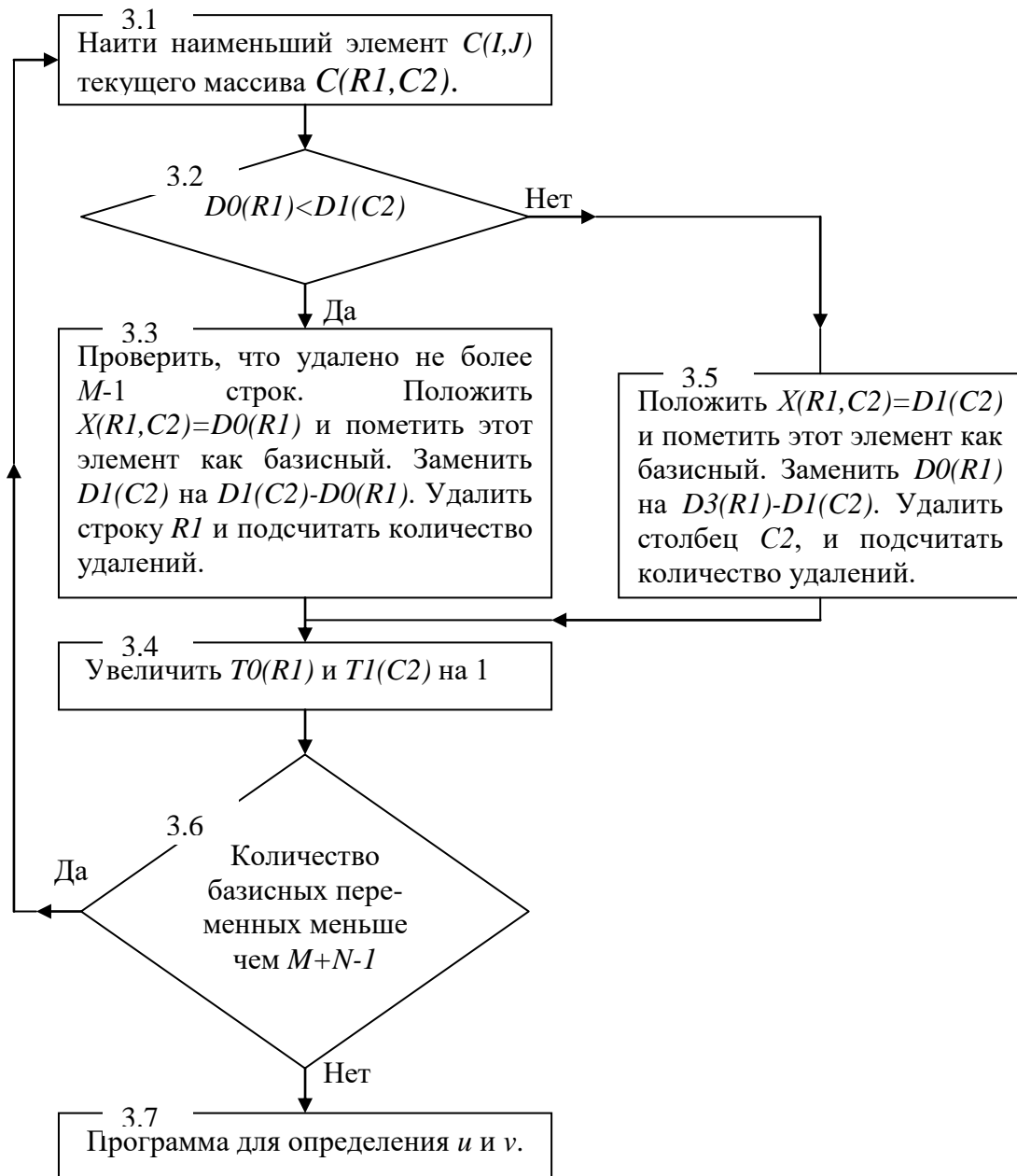


Рис. 2.11

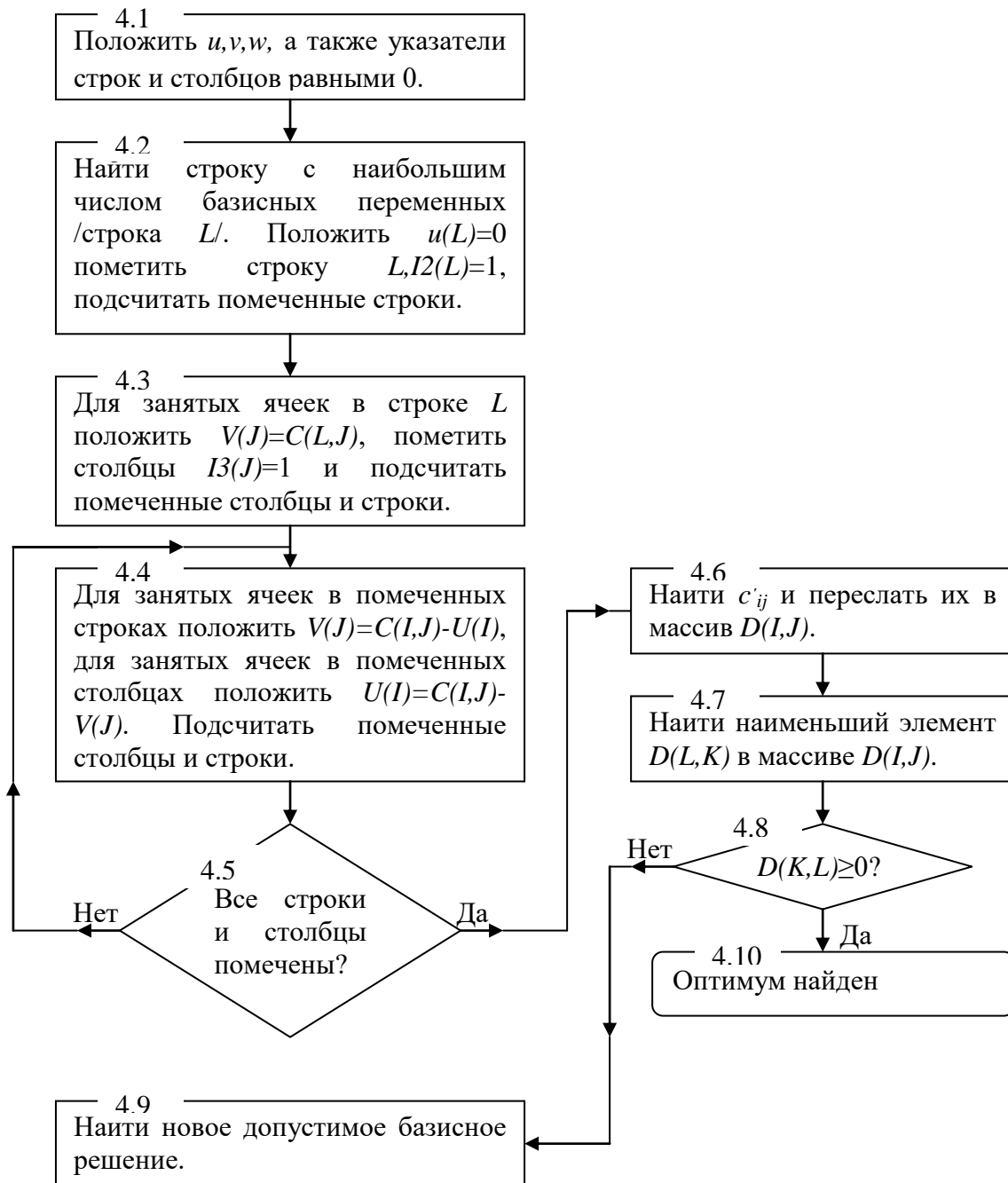


Рис. 2.12

3. ПОСЛЕДОВАТЕЛЬНОСТЬ ВЫПОЛНЕНИЯ РАБОТЫ

1. Ознакомится с теоретическими положениями (гл. 2).
2. Ознакомится с работой программы нахождения оптимального решения транспортной задачи на контрольном примере (приложения 1 и 2).
3. Получить у преподавателя индивидуальное задание (гл. 6).
4. В соответствии с заданием, внося необходимые изменения в программу (приложения 1 и 2), решить транспортную задачу.
5. Вывести полученный результат на печать.

4. СОДЕРЖАНИЕ ОТЧЁТА

1. Исходное задание.
2. Распечатка полученного результата.

5. ТЕХНИКА БЕЗОПАСНОСТИ

1. При выполнении лабораторной работы необходимо соблюдать требования техники безопасности, распространённые на работы с электрическими приборами с напряжением до 220 В.
2. В случае обнаружения неисправности ПЭВМ необходимо сообщить об этом преподавателю.

6. ПРИМЕРЫ ЗАДАНИЙ

1. Решите следующую транспортную задачу (рис. 6.1):

4	3	3	1	8
3	2	4	8	11
5	4	6	3	16
4	9	9	13	

Рис. 6.1

2. Решите следующую транспортную задачу (рис. 6.2):

2	9	3	10	8	7	9
3	2	6	3	5	2	14
1	8	2	4	1	5	16
4	8	7	6	6	6	11
6	4	10	13	7	10	

Рис. 6.2

3. Сталеплавильная компания располагает тремя заводами M_1 , M_2 , M_3 , способными произвести за некоторый промежуток времени 50, 30, 20 тыс. т стали соответственно. Свою продукцию компания поставляет четырём потребителям C_1 , C_2 , C_3 и C_4 , потребности которых составляют соответственно 12, 15, 25 и 36 тыс. т стали. Стоимости производств и транспортировки 1 тыс. т стали с различных заводов различным потребителям приведены в табл. 6.1:

Таблица 6.1

Потребитель	Завод		
	M_1	M_2	M_3
C_1	15	19	14
C_2	19	18	16
C_3	19	18	20
C_4	15	19	18

Определите минимальные общую стоимость, объемы производств на каждом заводе и планы перевозок.

4. Компания контролирует три фабрики F_1 , F_2 , F_3 , способные произвести 50, 25 и 25 тыс. изделий еженедельно. Она заключила договоры с четырьмя заказчиками C_1 , C_2 , C_3 и C_4 , которым требуется еженедельно 15, 20, 25 и 30 тыс. изделий. Стоимости производства и транспортировки 1 тыс. изделий заказчиками с фабрики приведены в табл. 6.2:

Таблица 6.2

Фабрика	Заказчик			
	C_1	C_2	C_3	C_4
F_1	13	17	17	14
F_2	18	16	16	18
F_3	12	14	19	17

Определите минимизирующие общую стоимость, объёмы производства и распределения для каждой фабрики.

5. Компания владеет двумя фабриками F_1 и F_2 , производящими оборудование. Фабрики в течение некоторого периода выпускают 16 и 12 тыс. изделий соответственно при нормальных темпах производства. При сверхурочной работе эти показатели могут быть повышены соответственно до 20 и 14 тыс. изделий. Дополнительная стоимость производства 1000 изделий в сверхурочное время на F_1 и на F_2 составляет 8 единиц. Компания снабжает трёх потребителей C_1 , C_2 и C_3 , потребности которых в течение одного и того же периода составляют соответственно 10, 13 и 7 тыс. изделий. Стоимость перевозок 1 тыс. изделий потребителю с фабрик приведены в табл. 6.3:

Таблица 6.3

Фабрика	Потребитель		
	C_1	C_2	C_3
F_1	5	4	6
F_2	6	3	2

Сформулируйте задачу нахождения оптимальных планов производства и распределения как транспортную и найдите её решение.

6. Четыре сталелитейных завода I, II, III и IV производят еженедельно соответственно 950, 300, 1350 и 450 т стали определённого сорта. Стальные болванки должны быть переданы потребителям A, B, C, D, E, еженедельные запасы которых составляют соответственно 250, 1000, 700, 650 и 450 т стали.

Стоимость транспортировки от заводов к потребителям в тоннах приведена в табл. 6.4:

Таблица 6.4

Завод	Потребитель				
	A	B	C	D	E
1	12	16	21	19	32
2	4	4	9	5	24
3	3	8	14	10	26
4	24	33	36	34	49

Какой нужно составить план распределения стальных болванок, чтобы минимизировать общую стоимость.

7. Компания владеет тремя заводами A, B, C. Соответствующие стоимости производства равны 26, 23 и 22 на единицу, объём производства 6000, 3000 и 3000 единиц. Компания обязалась поставлять соответственно

1500, 2500, 2700 и 3300 единиц в города W, X, Y, Z. При заданных стоимостях перевозок составьте оптимальные планы производства и распределения (табл. 6.5).

Таблица 6.5

Город	Стоимость транспортировки,		
	A	B	C
W	1	9	6
X	4	2	1
Y	1	2	7
Z	9	8	3

БИБЛИОГРАФИЧЕСКИЙ СПИСОК

1. Белолипецкий А. А. Экономико-математические методы : учеб. для вузов по спец. "Экономика" / А. А. Белолипецкий, В. А. Горелик. - М. : Академия, 2010. - 362, [2] с.
2. Банди Б. Основы линейного программирования: Пер.с англ. – М.: Радио и связь, 1989. – 176 с.
3. Геронимус Б.Л., Царфин Л.В. Экономико-математические методы в планировании на автомобильном транспорте. – М.: Транспорт, 1988. – 192 с.

Приложение 1

```
20 PRINT"                РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ"
21 LPRINT"                РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ":LPRINT
30 REMARK В ЗАДАЧЕ РАССМАТРИВАЕТСЯ М СТРОК И N СТОЛБЦОВ
35 DATA 3,5
40 READ M,N
60 DIM A(10),D0(10),B(10),D1(10),10(10),11(10),T0(10),T1(10)
70 DIM U(10),V(10),12(10),13(10)
80 DIM R0(30),C3(20)
100 DIM C(10,10),X(10,10),14(10,10),D(10,10),M0(10,10),C1(10)
140 REMARK ВВОД СТОИМОСТЕЙ, ОБЩИХ СТРОК И СТОЛБЦОВ
150 FOR I=1 TO M
160 FOR J=1 TO N
162 DATA 1,0,3,4,2
164 DATA 5,1,2,3,3
166 DATA 4,8,1,4,3
170 READ C(I,J)
171 LET 14(I,J)=0:LET 10(I)=0:LET 11(J)=0:LET R0(I)=0
173 LET T1(J)=0:LET T0(I)=0:LET X(I,J)=0
180 NEXT J
190 NEXT I
200 FOR I=1 TO M
201 DATA 15,25,20
202 READ A(I)
204 LET D0(I)=A(I)
206 NEXT I
210 FOR J=1 TO N
211 DATA 20,12,5,8,15
212 READ B(J)
214 LET D1(J)=B(J)
216 NEXT J
500 LET C=0
502 LET C3=0
504 LET C1=0
510 LET R1=0
512 LET C2=0
514 LET Y=1E10
600 FOR I=1 TO M
610 IF 10(I)=1 THEN GOTO 670
620 FOR J=1 TO N
630 IF 11(J)=1 THEN GOTO 660
640 IF C(I,J)>Y THEN GOTO 660
650 LET Y=C(I,J)
652 LET R1=I
654 LET C2=J
660 NEXT J
670 NEXT I
700 IF D0(R1)<=D1(C2) THEN GOTO 760
710 LET X(R1,C2)=R1(C2)
720 LET 14(R1,C2)=1
```


Продолжение приложения 1

```
730 LET D0(R1)=D0(R1)-D1(C2)
732 LET D1(C2)=0
740 LET 11(C2)=1
742 LET C0=C0+1
744 LET C3=C3+1
750 GOTO 810
760 IF D0(R1)=D1(C2) THEN IF C1=M-1 THEN 710
770 LET X(R1,C2)=D0(R1)
780 LET 14(R1,C2)=1
790 LET D1 (C2)=D1(C2)-D0(R1)
792 LET D0(R1)=0
800 LET 10(R1)=1
802 LET C0=C0+1
804 LET C1=C1+1
810 LET T0(R1)=T0(R1)+1
812 LET T1(C2)=T1(C2)+1
820 IF C0<M+N-1
1000 FOR I=1 TO M
1010 LET 12(I)=0
LET U(I)=0
1020 NEXT I
1030 FOR J=1 TO N
1040 LET 13(J)=0
1042 LET V(J)=0
1050 NEXT J
1080 LET T=0
1082 LET L=0
1100 FOR I=1 TO M
1110 IF T0(I)<T THEN GOTO 1130
1120 LET T=T0(I)
1122 LET L=I
1130 NEXT I
1140 LET U(L)=0
1142 LET 12(L)=1
1144 LET C0=1
1146 LET C1=1
1148 LET C3=0
1150 FOR J=1 TO N
1160 IF 14(L,J)=0 THEN GOTO 1190
1170 LET V(J)=C(L,J)
1172 LET 13(J)=1
1180 LET C3=C3+1
1182 LET C0=C0+1
1190 NEXT J
1197 NEXT I
1200 FOR I=1 TO M
1210 FOR J=1 TO N
1220 IF 14(I,J)=0 THEN GOTO 1310
1230 IF 12(I)=0 THEN IF 13(J)=0 THEN 1310
```

Продолжение приложения 1

```
1240 IF 12(I)=1 THEN IF 13(J)=1 THEN 1310
1250 IF 12(I)=0 THEN IF 13(J)=1 THEN 1290
1260 LET V(J)=C(I,J)-U(I)
1262 LET 13(J)=1
1270 LET C3=C3+1
1272 LET C0=C0+1
1280 GOTO 1310
1290 LET U(I)=C(I,J)-V(J)
1292 LET 12(I)=1
1300 LET C1=C1+1
1302 LET C0=C0+1
1310 NEXT J
1320 NEXT I
1330 IF C0<>M+N THEN GOTO 1200
1332 PRINT"ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ"
1333 LPRINT"ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ"
1334 PRINT"U(I)",
1335 LPRINT"U(I)",
1336 FOR I=1 TO M
1337 PRINT U(I);LPRINT U(I);
1338 NEXT I
1339 PRINT:LPRINT
1340 PRINT"V(J)";
1341 LPRINT"V(J)";
1342 FOR J=1 TO N
1344 PRINT V(J);
1345 LPRINT V(J);
1346 NEXT J
1348 PRINT
1349 LPRINT
1400 FOR I=1 TO M
1410 FOR J=1 TO N
1420 IF 14(I,J)=0 THEN GOTO 1460
1430 LET D(I,J)=C(I,J)-U(I)-V(J)
1440 IF D(I,J)<>0 THEN PRINT"ОШИБКА 1"
1450 GOTO 1470
1460 LET D(I,J)=C(I,J)-U(I)-V(J)
1470 NEXT J
1480 NEXT I
1500 LET T=0
1502 LET K=0
1504 LET L=0
1510 FOR I=1 TO M
1520 FOR J=1 TO N
1530 IF 14(I,J)=1 THEN GOTO 1560
1540 IF D(I,J)>=T THEN GOTO 1560
1550 LET T=D(I,J)
1552 LET K=I
1554 LET L=J
```

```
1560 NEXT J
1570 NEXT I
1580 IF T=0 THEN GOTO 4000
1581 PRINT:LPRINT
1582 PRINT"C'KL="T;:LPRINT"C'KL="T,
1583 PRINT"K="K;:LPRINT"K="K;
1584 PRINT"L="L
1585 LPRINT"L="L
1586 PRINT:LPRINT
1587 GOSUB 5000
2000 FOR I=1 TO M
2001 LET 12(I)=0
2002 NEXT I
2010 FOR J=1 TO N
2011 LET 13(J)=0
2012 NEXT J
2015 FOR I=1 TO M+N
2016 LET R0(I)=0
2017 LET C3(I)=0
2018 NEXT I
2020 FOR I=1 TO M
2021 FOR J=1 TO N
2030 LET D(I,J)=0
2035 LET M0(I,J)=0
2040 NEXT J
2045 NEXT I
2050 LET T=1
2055 LET 15=0
2060 LET R0(T)=K
2062 LET C3(T)=L
2070 LET D(K,L)=1
2071 LET M0(K,L)=1
2072 LET 12(K)=1
2073 PRINT T,K;L
2074 LPRINT T,K;L
2100 LET F0=0
2101 LET F1=0
2102 LET R1=R0(T)
2103 LET C2=0
2110 FOR J=1 TO N
2120 IF=1 THEN GOTO 2180
2130 IF 14(R1,J)=0 THEN GOTO 2180
2140 IF 13(J)=1 THEN GOTO 2180
2150 IF M0(R1,J)=1 THEN GOTO 2180
2155 IF T1(J)=1 THEN IF J=L THEN GOTO 2170
2160 IF T1(J)=1 THEN LET 15=1:GOTO 2180
2170 LET F1=1
2171 LET C2=J
2172 LET 13(J)=1
```

Продолжение приложения 1

```
2173 LET J=N
2180 NEXT J
2190 IF C2<>0 THEN GOTO 2300
2200 IF 15>0 THEN LET 15=0
2210 LET D(R0(T),C3(T))=0
2215 LET T=T-1
2220 GOTO 2500
2300 LET T=T+1
2310 LET R0(T)=R1
2315 LET C3(T)=C2
2320 LET D(R1,C2)=-1
2325 LET M0(R1,C2)=-1
2327 PRINT T,R1;C2
2328 LPRINT T,R1;C2
2400 IF C3(T)=L THEN IF T>2 THEN GOTO 3000
2500 LET F0=0
2501 LET F1=0
2502 LET R1=0
2503 LET C2=C3(T)
2510 FOR I=1 TO M
2520 IF F0=1 THEN GOTO 2580
2530 IF 14(I,C2)=0 THEN GOTO 2580
2540 IF 12(I)=1 THEN GOTO 2580
2550 IF M0(I,C2)=1 THEN GOTO 2580
2560 IF T0(I)=1 THEN IF 15=0 THEN LET 15=1:GOTO 2580
2570 LET F0=1
2571 LET R1=1
2573 LET 12(I)=1
2575 LET I=M
2580 NEXT I
2590 IF R1<>0 THEN GOTO 2700
2600 IF 15>0 THEN LET 15=0
2610 LET D(R0(T),C3(T))=0
2615 LET T=T-1
2620 GOTO 2100
2700 LET T=T+1
2705 LET 15=0
2710 LET R0(T)=R1
2715 LET C3(T)=C2
2720 LET D(R0(T),C3(T))=1
2721 LET M0(R1,C2)=1
2725 PRINT T,R1;C2
2726 LPRINT T,R1;C2
2730 GOTO 2100
3000 LET W=1E10
3004 LET K0=0
3010 FOR I=2 TO T STEP 2
3020 IF X(R0(I),C3(I))>=W THEN GOTO 3040
3030 LET W=X(R0(I),C3(I)).
```

Продолжение приложения 1

```
3035 LET K0=R0(I)
3037 LET L0=C3(I)
3040 NEXT I
3100 FOR I=1 TO T
3110 LET X(R0(I),C3(I))=X(R0(I),C3(I))+W*D(R0(I),C3(I))
3120 NEXT I
3200 LET 14(K,L)=1
3205 LET 14(K0,L0)=0
3210 LET T0(K)=T0(K)+1
3215 LET T0(K0)=T0(K0)-1
3220 LET T1(L)=T1(L)+1
3225 LET T1(L0)=T1(L0)-1
3227 PRINT "W="W;:LPRINT"W="W;
3228 PRINT"K0="K0;:LPRINT"K0="K0;
3229 PRINT "L0="L0:LPRINT"L0="L0
3230 PRINT"ПРЕОБРАЗОВАНИЕ ЗАКОНЧЕНО УСПЕШНО"
3235 LPRINT"ПРЕОБРАЗОВАНИЕ ЗАКОНЧЕНО УСПЕШНО"
3250 GOTO 1000
4000 PRINT"ОКОНЧАТЕЛЬНОЕ РЕШЕНИЕ"
4005 LPRINT"ОКОНЧАТЕЛЬНОЕ РЕШЕНИЕ"
4010 GOSUB 5000
4500 END
5000 LET C4=0
5010 PRINT " I J   XJ   CIL   СТОИМОСТЬ"
5015 LPRINT " I J   XJ   CIL   СТОИМОСТЬ"
5020 FOR I=1 TO M
5030 FOR J=1 TO N
5040 IF 14(I,J)=0 THEN GOTO 5110
5050 LET P0=C(I,J)*X(I,J)
5060 LET C4+P0
5070 PRINT I;J;
5075 LPRINT I;J;
5080 LET P1=X(I,J)
5082 PRINT"    "P1;
5083 LPRINT"    "P1;
5084 LET P1=C(I,J)
5086 PRINT"    "P1;
5090 LPRINT"    "P1;
5094 LET P1=P0
5096 PRINT"    "P1;
5098 LPRINT"    "P1;
5100 PRINT
5105 LPRINT
5110 NEXT J
5115 NEXT I
5200 PRINT"ОБЩАЯ СТОИМОСТЬ РАВНА ";C4
5205 LPRINT"ОБЩАЯ СТОИМОСТЬ РАВНА ";C4
5250 RETURN
6000 STOP
```

РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ

ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ

U(I) -4 0 0

V(J) 5 4 1 3 3

C'KL= -3 K= 2 L= 2

I	J	KIJ	CIJ	СТОИМОСТЬ
1	1	3	1	3
1	2	12	0	0
2	1	17	5	85
2	4	8	3	24
2	5	0	3	0
3	3	5	1	5
3	5	15	3	45

ОБЩАЯ СТОИМОСТЬ РАВНА 162

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

W= 12 K0= 1 L0= 2

ПРЕОБРАЗОВАНИЕ ОКОНЧЕНО УСПЕШНО

ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ

U(I) -4 0 0

V(J) 5 1 1 3 3

C'KL= -1 K= 3 L= 1

I	J	KIJ	CIJ	СТОИМОСТЬ
1	1	15	1	15
2	1	5	5	25
2	0	12	1	15
2	4	8	3	24
2	5	0	3	0
3	3	5	1	5
3	5	15	3	45

ОБЩАЯ СТОИМОСТЬ РАВНА 126

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

W= 5 K0= 2 L0= 1

ПРЕОБРАЗОВАНИЕ ОКОНЧЕНО УСПЕШНО

ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ

U(I) -3 0 0

V(J) 4 1 1 3 3

ОКОНЧАТЕЛЬНОЕ РЕШЕНИЕ

I	J	KIJ	CIJ	СТОИМОСТЬ
---	---	-----	-----	-----------

1	1	15	1	15
---	---	----	---	----

2	2	12	1	12
---	---	----	---	----

2	4	8	3	24
---	---	---	---	----

2	5	5	3	15
---	---	---	---	----

3	1	5	4	20
---	---	---	---	----

3	3	5	1	5
---	---	---	---	---

3	5	10	3	30
---	---	----	---	----

ОБЩАЯ СТОИМОСТЬ РАВНА 121

```

uses Crt,Graph;
const
  inter:set of char=['0','1','2','3','4','5','6','7','8','9'];
  inter1:set of char=['0','1','2','3','4','5','6','7','8','9'];
  SetYN:set of char=['y','Y','h','H','n','N','T','t'];
  Cconst:array[1..3,1..5] of integer=
    ((1,0,3,4,2),(5,1,2,3,3),(4,8,1,4,3));
  Aconst:array[1..3] of integer=(15,25,20);
  Bconst:array[1..5] of integer=(20,12,5,8,15);
var
  ch:char;
  grDriver,grMode,ErrCode: Integer;
  M,N:integer;
  X:array[1..10,1..10] of real;
  Cm,I4,D,M0:array[1..10,1..10] of integer;
  A,D0,B,D1,I0,I1,T0,T1,U,V,I2,I3,C1m:array[1..10] of integer;
  R0:array[1..30] of integer;
  C3m:array[1..20] of integer;
  C,C3,C1,R1,C2,C0:integer;
  Y,W,C4:real;
  i,j,t,l,l0,k,k0,i5,f0,f1:integer;
  ind,Balans:string;
  IndZik:integer;

procedure InInt1( var k:integer);
var
  ch:char;
  St:string;
  i,code:word;
begin
  St:="";
  i:=1;
  repeat
    ch:=readkey;
    while (not(ch in inter1))and(ch<>#13)and(ch<>#8) do ch:=readkey;
    if (ch<>#13)and(ch<>#8)and(i<=1) then
      begin
        write(ch);
        St:=St+ch;
        i:=i+1;
      end;
  until (ch=#8)and(St<>") then
  begin
    Delete(St,length(St),1);
    GotoXY(WhereX-1,WhereY);
    write(' ');
    GotoXY(WhereX-1,WhereY);
    i:=i-1;
  end;
end;

```


Продолжение приложения 2

```
    end;  
    val(St,k,code);  
until (ch=#13)and(St<>"")and(k<>0);  
val(St,k,code);  
end;
```

```
procedure InInt4( var k:integer);  
var  
    ch:char;  
    St:string;  
    i,code:word;  
begin  
    St:="";  
    i:=1;  
    repeat  
        ch:=readkey;  
        while (not(ch in inter1))and(ch<>#13)and(ch<>#8) do ch:=readkey;  
        if (ch<>#13)and(ch<>#8)and(i<=4) then  
            begin  
                write(ch);  
                St:=St+ch;  
                i:=i+1;  
            end;  
        if (ch=#8)and(St<>"") then  
            begin  
                Delete(St,length(St),1);  
                GotoXY(WhereX-1,WhereY);  
                write(' ');  
                GotoXY(WhereX-1,WhereY);  
                i:=i-1;  
            end;  
        val(St,k,code);  
    until (ch=#13)and(St<>"")and(k<>0);  
    val(St,k,code);  
end;
```

```
procedure InIntegerT( var k:integer);  
var  
    ch:char;  
    St:string;  
    i,code:word;  
begin  
    St:="";  
    i:=1;  
    repeat  
        ch:=readkey;  
        while (not(ch in inter))and(ch<>#13)and(ch<>#8) do ch:=readkey;
```

```
if (ch<>#13)and(ch<>#8)and(i<=2) then
begin
  write(ch);
  St:=St+ch;
  i:=i+1;
end;
if (ch=#8)and(St<>") then
begin
  Delete(St,length(St),1);
  GotoXY(WhereX-1,WhereY);
  write(' ');
  GotoXY(WhereX-1,WhereY);
  i:=i-1;
end;
until (ch=#13)and(St<>");
val(St,k,code);
end;

procedure NLK;
begin
  TextColor(2);
  writeln;
  write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
  ch:=readkey;
  TextColor(15);
end;

procedure Color(X,Y:integer);
begin
  TextColor(X);
  TextBackGround(Y);
end;

procedure BarR(Color,X,Y,a,b:integer);
var i,j:word;
begin
  TextColor(Color);
  GoToXY(X,Y);
  write('r');
  for i:=0 to a-2 do write('=');
  writeln('¬');
  for j:=0 to b-2 do
begin
  GoToXY(X,WhereY);
  write('|');
  GoToXY(X+a,WhereY);
  writeln('|');
end;
GoToXY(X,WhereY);
```



```

delta:=trunc(42/N);
for i:=1 to N-1 do
begin
  GoToXY(9+i*delta,4); write('┐');
  GoToXY(9+i*delta,5); write('└');
  GoToXY(9+i*delta,6); write('┌');
end;
GoToXY(12,5);
for i:=1 to N do
begin
  write(i);
  GoToXY(12+i*delta,5);
end;
GoToXY(1,7);
for i:=1 to M do
begin
  write(' | 'i,' | ');
  for j:=1 to N-1 do begin
    GoToXY(9+j*delta,WhereY);
    write(' | ');
  end;
  GoToXY(51,WhereY); write(' | ');
  writeln;
end;
write(' ────────────');
for j:=1 to N-1 do begin
  GoToXY(9+j*delta,WhereY);
  write('└');
end;
GoToXY(51,WhereY); write('└');
TextColor(10);
for i:=1 to M do
begin
  GoToXY(12,7+(i-1));
  for j:=1 to N do
  begin
    write(Cm[i,j]);
    GoToXY(12+j*delta,WhereY);
  end;
  writeln;
end;
Xstr:=12+N*delta+2;
{возможность изменения данных-стоимостей перевозок}
GoToXY(1,22);
TextColor(15);write('СТРЕЛКИ');
TextColor(10);write('-переход ');
TextColor(15);write('ЗАБОЙ,ENTER');
TextColor(10);write('-ввод данных ');
TextColor(15);write('ПРОБЕЛ');

```

```
TextColor(10);write('-продолжение работы');
GoToXY(12,7);
xx:=1;yy:=1;
Wiw(15);
repeat
ch:=readkey;
case ch of
  #80{вправо}: if xx<M then
    begin
      Wiw(10);
      xx:=xx+1;
      Wiw(15);
    end;
  #72{влево} : if xx>1 then
    begin
      Wiw(10);
      xx:=xx-1;
      Wiw(15);
    end;
  #77{вниз} : if yy<N then
    begin
      Wiw(10);
      yy:=yy+1;
      Wiw(15);
    end;
  #75{вверх} : if yy>1 then
    begin
      Wiw(10);
      yy:=yy-1;
      Wiw(15);
    end;
  #8{ввод} : begin
    TextColor(15);
    GoToXY(12+(yy-1)*delta,7+(xx-1));
    write(' ');
    GoToXY(12+(yy-1)*delta,7+(xx-1));
    InIntegerT(Cm[xx,yy]);
    GoToXY(12+(yy-1)*delta,7+(xx-1));
    write(' ');
    Wiw(10);
    if yy<>N then yy:=yy+1 else yy:=yy-1;
    Wiw(15);
  end;
end; {case}
until ch=#32; {пробел}
GoToXY(1,22); ClrEol;
if (M=3)and(N=5) then
begin
  TextColor(14);
```

Продолжение приложения 2

```
GoToXY(1,12+M);
writeln(* ДЛЯ РАБОТЫ С ВСТРОЕННЫМИ A[i] и B[j] НАЖМИТЕ КЛАВИШУ
"Y");
writeln(* ДЛЯ ВВОДА A[i] и B[j] НАЖМИТЕ ЛЮБУЮ КЛАВИШУ);
ch:=readkey;
TextColor(0);
GoToXY(1,12+M); ClrEol;
GoToXY(1,12+M+1); clrEol;
end;
IF (ch<>'Y')and(ch<>'y')and(ch<>'н')and(ch<>'Н') then
BEGIN
  {ввод данных-общих строк}
  GoToXY(12,12+M);
  TextColor(10);
  writeln('ВВОД ОБЩИХ СТРОК ');
  for i:=1 to M do
  begin
    writeln;
    BarR(10,5,12+M+1,15,2);
    GoToXY(6,12+M+2);writeln('      ');
    GoToXY(6,12+M+2);
    TextColor(10);
    write(' A',i,'= ');
    TextColor(15);
    InInt4(A[i]);
    GoToXY(Xstr,7+(i-1));
    write(A[i]);
    D0[i]:=A[i];
  end;
  TextColor(0); {забой}
  GoToXY(12,12+M);
  writeln('ВВОД ОБЩИХ СТРОК ');
  writeln; BarR(0,5,12+M+1,15,2); TextColor(0);
  GoToXY(6,12+M+2);write(' A',M,'= ',A[M]);
  Sound(200);
  delay(500);
  NoSound; delay(1000);

  {ввод данных-общих столбцов}
  GoToXY(12,12+M);
  TextColor(10);
  writeln('ВВОД ОБЩИХ СТОЛБЦОВ ');
  for i:=1 to N do
  begin
    writeln;
    BarR(10,5,12+M+1,15,2);
    GoToXY(6,12+M+2);writeln('      ');
    GoToXY(6,12+M+2);
    TextColor(10);
```

```
write(' B',i,'= ');
TextColor(15);
InInt4(B[i]);
GoToXY(12+(i-1)*delta,7+M+1);
write(B[i]);
D1[i]:=B[i];
end;
TextColor(0); {забой}
GoToXY(12,12+M);
writeln('ВВОД ОБЩИХ СТОЛБЦОВ ');
writeln;
BarR(0,5,12+M+1,15,2);
GoToXY(6,12+M+2);
TextColor(0);
write(' B',N,'= ');
TextColor(0);
write(B[N]);
GoToXY(1,WhereY);
TextColor(10);
write(' НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
TextColor(15);
END
ELSE {вывод встроенных А и В}
BEGIN
  TextColor(15);
  for i:=1 to M do
    begin
      GoToXY(Xstr,7+(i-1));
      write(A[i]);
    end;
  for i:=1 to N do
    begin
      GoToXY(12+(i-1)*delta,7+M+1);
      write(B[i]);
    end;
  GoToXY(2,7+M+4);
  TextColor(10);
  write(' НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
END;
ch:=readkey;
end;

Procedure SubSub;
begin {2100}
REPEAT
  f0:=0; f1:=0;
  R1:=R0[t];
  C2:=0;
  for j:=1 to N do
```

Продолжение приложения 2

```
if (f1<>1)and(I4[R1,j]<>0)and(I3[j]<>1)and(M0[R1,j]<>1) then
begin
  if (T1[j]=1)and(j<>1) then i5:=1
    else
      begin
        f1:=1;
        C2:=j;
        I3[j]:=1;
        j:=N
      end;
end;
if C2<>0 then
begin
  t:=t+1;
  R0[t]:=R1;
  C3m[t]:=C2;
  D[R1,C2]:=-1;
  M0[R1,C2]:=1;
  writeln(' ',t,' ',R1,' ',C2);
  if (C3m[t]=1)and(t>2) then exit;
end else
begin
  if I5>0 then I5:=0;
  D[R0[t],C3m[t]]:=0;
  t:=t-1
end;
f0:=0;
f1:=0;
R1:=0;
C2:=C3m[t];
for i:=1 to M do
if (f0<>1)and(I4[i,C2]<>0)and(I2[i]<>1)and(M0[i,C2]<>1)then
  if (T0[i]=1)and(I5=0) then I5:=1
    else begin
      f0:=1;
      R1:=i;
      I2[i]:=1;
      i:=M;
    end;
if R1=0 then begin
  if I5>0 then I5:=0;
  D[R0[t],C3m[t]]:=0;
  t:=t-1;
end else
begin
  t:=t+1;
  I5:=0;
  R0[t]:=R1;
  C3m[t]:=C2;
```



```

D[R0[t],C3m[t]]:=1;
M0[R1,C2]:=1;
writeln(' ',t,' ',R1,' ',C2);
end;
UNTIL false;
end;

```

Procedure OutResult;

```
var P0:real;
```

```
procedure tabl;
```

```
begin
```

```
  TextColor(2);
```

```
  writeln(' ');
```

```
  writeln(' ');
```

```
  writeln(' ');
```

```
end;
```

```
begin {5000}
```

```
  c4:=0;
```

```
  tabl;
```

```
  for i:=1 to M do
```

```
    for j:=1 to N do
```

```
      if I4[i,j]<>0 then
```

```
        begin
```

```
          P0:=Cm[i,j]*X[i,j];
```

```
          C4:=C4+P0;
```

```
          TextColor(2);
```

```
          write(' ');
```

```
          TextColor(15);
```

```
          GoToXY(7,WhereY); write(i);
```

```
          GoToXY(14,WhereY); write(j);
```

```
          GoToXY(20,WhereY); write(X[i,j]:5:1); {!!!!!!!!!!!!!!}
```

```
          GoToXY(30,WhereY); write(Cm[i,j]);
```

```
          GoToXY(40,WhereY); write(P0:5:1);
```

```
          writeln;
```

```
        end;
```

```
  TextColor(2);
```

```
  writeln(' ');
```

```
  TextColor(10);
```

```
  write(' ОБЩАЯ СТОИМОСТЬ =');
```

```
  TextColor(15); write(C4:6:1);
```

```
  TextColor(10);
```

```
  writeln;
```

```
  GoToXY(5,WhereY); write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
```

```
  ch:=readkey;
```

```
  TextColor(0); GoToXY(5,WhereY); write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
```

```
  TextColor(15);
```

```
end;
```

Procedure SubPr;

```
begin {1580}
  ind:="";
  if t=0 then begin ind:='exit'; exit end;
  TextColor(10);
  write(' C[K,L]=');
  TextColor(15); write(t);
  TextColor(10); write(' K=');
  TextColor(15); write(k);
  TextColor(10); write(' L=');
  TextColor(15); writeln(l);
  OutResult;
  for i:=1 to M do I2[i]:=0;
  for j:=1 to N do I3[j]:=0;
  for i:=1 to M+N do
    begin
      R0[i]:=0;
      C3m[i]:=0
    end;
  for i:=1 to M do
    for j:=1 to N do
      begin
        D[i,j]:=0;
        M0[i,j]:=0
      end;
  t:=1;
  i5:=0;
  R0[t]:=k; C3m[t]:=l; D[k,l]:=1; M0[k,l]:=1;
  I2[k]:=1;
  GoToXY(1,WhereY); ClrEol;
  writeln(' ',t,' ',k,' ',l);
  SubSub;
  W:=1E10;
  k0:=0;
  i:=2;
  while i<=t do
    begin
      if X[R0[i],C3m[i]]<W then
        begin
          W:=X[R0[i],C3m[i]];
          k0:=R0[i];
          l0:=C3m[i]
        end;
      i:=i+2;
    end;
  for i:=1 to t do
    X[R0[i],C3m[i]]:=X[R0[i],C3m[i]]+W*D[R0[i],C3m[i]];
  I4[k,l]:=1;
  I4[k0,l0]:=0;
  T0[k]:=T0[k]+1;
```

Продолжение приложения 2

```
T0[k0]:=T0[k0]-1;
T1[l]:=T1[l]+1;
T1[l0]:=T1[l0]-1;
TextColor(10); write(' W=');
TextColor(15); write(W:5:3);
TextColor(10); write(' k0=');
TextColor(15); write(k0);
TextColor(10); write(' l0=');
TextColor(15); writeln(l0);
TextColor(14);
writeln(' ПРЕОБРАЗОВАНИЕ ЗАКОНЧЕНО УСПЕШНО');
TextColor(10);
GoToXY(5,WhereY); write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
ch:=readkey;
TextColor(0); GoToXY(5,WhereY); write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
TextColor(15);
ClrScr;
end;
```

Procedure Rash;

Procedure Uslow;

```
begin
  if (I4[i,j]=0) then exit;
  if (I2[i]=0) then if (I3[j]=0) then exit;
  if (I2[i]=1) then if (I3[j]=1) then exit;
  if (I2[i]=0)and(I3[j]=1) then
    begin
      U[i]:=Cm[i,j]-V[j];
      I2[i]:=1;
      C1:=C1+1;
      C0:=C0+1
    end else
      begin
        V[j]:=Cm[i,j]-U[i];
        I3[j]:=1;
        C3:=C3+1;
        C0:=C0+1
      end;
end;
```

end;

begin

REPEAT

R1:=0;

C2:=0;

Y:=1E10;

for i:=1 to M do

if I0[i]<>1 then

for j:=1 to N do

if (I1[j]<>1)and(Cm[i,j]<=Y) then

```
begin
  Y:=Cm[i,j];
  R1:=i;
  C2:=j
end;
if ((D0[R1]=D1[C2])and(C1=M-1))or(D0[R1]>D1[C2]) then
begin
  X[R1,C2]:=D1[C2];
  I4[R1,C2]:=1;
  D0[R1]:=D0[R1]-D1[C2];
  D1[C2]:=0;
  I1[C2]:=1;
  C0:=C0+1;
  C3:=C3+1
end else
begin
  X[R1,C2]:=D0[R1];
  I4[R1,C2]:=1;
  D1[C2]:=D1[C2]-D0[R1];
  D0[R1]:=0;
  I0[R1]:=1;
  C0:=C0+1;
  C1:=C1+1
end;
T0[R1]:=T0[R1]+1;
T1[C2]:=T1[C2]+1;
UNTIL (C0>=M+N-1);
C1:=C1+1;
REPEAT
for i:=1 to M do
begin
  I2[i]:=0;
  U[i]:=0;
end;
for j:=1 to N do
begin
  I3[j]:=0;
  V[j]:=0;
end;
t:=0;
l:=0;
for i:=1 to M do
if T0[i]>=t then
begin
  t:=T0[i];
  l:=i
end;
U[l]:=0;
I2[l]:=1;
```

```
C0:=1;
C1:=1;
C3:=0;
for j:=1 to N do
  if I4[l,j]<>0 then
    begin
      V[j]:=Cm[l,j];
      I3[j]:=1;
      C3:=C3+1;
      C0:=C0+1
    end;
IndZik:=0;
REPEAT
  for i:=1 to M do
    for j:=1 to N do
      begin
        Uslow;
        IndZik:=IndZik+1;
      end;
UNTIL (C0=M+N)or(IndZik>10000);
if IndZik>10000 then
  begin
    GoToXY(1,10);
    TextColor(12);
    writeln('                ВНИМАНИЕ!');
    TextColor(14);
    writeln;
    writeln('        ПРОГРАММА НЕ МОЖЕТ ОБРАБОТАТЬ ИСХОДНЫЕ
ДАННЫЕ. ');
    writeln('        НЕОБХОДИМО ПРОВЕРИТЬ ВЫПОЛНЕНИЕ УСЛОВИЙ,
ВАЖНЫХ ');
    writeln('    ДЛЯ ДАННОЙ ЗАДАЧИ. ');
    Sound(400); delay(2000); NoSound;
    Sound(100); delay(2000); NoSound;
    ch:=readkey;
    exit;
  end;
TextColor(10);
writeln(' ДОПОЛНИТЕЛЬНЫЕ СТОИМОСТИ ');
write('  U[i] ');
TextColor(15);
for i:=1 to M do
  write(U[i], ' ');
writeln;
TextColor(10);
write('  V[j] ');
TextColor(15);
for j:=1 to N do
  write(V[j], ' ');
```

Продолжение приложения 2

```
writeln;
for i:=1 to M do
  for j:=1 to N do
    if I4[i,j]=0 then D[i,j]:=Cm[i,j]-U[i]-V[j]
    else
      begin
        D[i,j]:=Cm[i,j]-U[i]-V[j];
        if D[i,j]<>0 then begin
          TextColor(9);
          writeln(' ОШИБКА 1');
        end;
      end;
t:=0;
k:=0;
l:=0;
for i:=1 to M do
  for j:=1 to N do
    if (I4[i,j]<>1)and(D[i,j]<t) then
      begin
        t:=D[i,j];
        k:=i;
        l:=j;
      end;
SubPr;
if ind='exit' then exit;
UNTIL false
end;
```

Procedure start;

var

code:word;

yy:integer;

ind:boolean;

begin

repeat

ClrScr;

M:=3;

N:=5;

window(11,7,75,25);

TextColor(10);

writeln; writeln;

writeln(' ПРОГРАММА ');

writeln(' ДЛЯ РЕШЕНИЯ ТРАНСПОРТНОЙ ЗАДАЧИ');

BarR(10,5,1,45,5);

TextColor(15);

NLK;

GoToXY(1,WhereY);

ClrEol;

GoToXY(9,WhereY+3);

```
TextColor(10);
writeln('КОЛИЧЕСТВО СТРОК   M=3');
GotoXY(9,WhereY);
writeln('КОЛИЧЕСТВО СТОЛБЦОВ N=5');
BarR(2,5,WhereY-3,35,3);
GoToXY(1,18);
write('СТРЕЛКИ,ЗАБОЙ,ENTER');
TextColor(10);write('-изменение данных ');
TextColor(15);write('ПРОБЕЛ');
TextColor(10);write('-продолжение работы');
GoToXY(33,11); TextColor(15);
write(M);
yy:=11;
repeat
  ch:=readkey;
  if (ch=#72)or(ch=#80) then
    begin
      if yy=12 then
        begin
          yy:=11;
          GoToXY(33,12)
        end
      else begin
          yy:=12;
          GoToXY(33,11);
        end;
      TextColor(10);
      if yy=11 then write(N) else write(M);
      GoToXY(33,yy);
      TextColor(15);
      if yy=11 then write(M) else write(N);
    end;
  if ch=#8 then
    begin
      repeat
        ind:=true;
        TextColor(15);
        GoToXY(33,yy); write(' ');
        GotoXY(33,yy);
        if yy=11 then InInt1(M) else InInt1(N);
        if not((M>=2)and(M<=8)and(N>=2)and(N<=8))then
          begin
            ind:=false;
            GoToXY(10,15);
            write('НЕПРАВИЛЬНО ВВЕДЕНЫ ДАННЫЕ');
            Sound(700);
            delay(1000);
            NoSound;
            delay(6000);
```

```
        GoToXY(1,WhereY);
        ClrEol;
    end;
until ind;
    if yy=12 then
        begin
            yy:=11;
            GoToXY(33,12)
        end
    else begin
        yy:=12;
        GoToXY(33,11);
    end;
    TextColor(10);
    if yy=11 then write(N) else write(M);
    GoToXY(33,yy);
    TextColor(15);
    if yy=11 then write(M) else write(N);

    end;
until ch=#32;
TextColor(10);
GoToXY(33,yy); if yy=11 then write(M) else write(N);
GoToXY(1,18); ClrEol;
TextColor(15);
write('НАЖМИТЕ ЛЮБУЮ КЛАВИШУ');
ch:=readkey;
ClrScr;
window(1,1,70,25);
SetZn;
OutZn;
ClrScr;
Rash;
if IndZik<10000 then
BEGIN
    TextColor(15);
    writeln;
    writeln('        ОКОНЧАТЕЛЬНОЕ РЕШЕНИЕ');
    writeln('        =====');
    OutResult;
END;
ClrScr;
BarR(10,20,8,30,4);
GoToXY(27,10);
TextColor(15);
Write('ПОВТОРИТЬ? Д/Н');
ch:=readkey;
until (ch<>'Д')and(ch<>'д')and(ch<>'Н')and(ch<>'н');
end;

BEGIN
    Color(15,0);
    start;
END.
```


Учебно-методическое издание

РЕШЕНИЕ ТРАНСПОРТНОЙ ЗАДАЧИ НА ПЭВМ

Методические указания к практическим занятиям

Автор-составитель
ГуДИЛОВ Сергей Владимирович

Редактор

Лицензия

Подписано в печать

Формат 60х84 1/16.

Уч.-изд.л.

Тираж 100 экз.

Заказ №

Издательско-полиграфический центр

Новгородского государственного университета им. Ярослава Мудрого.
173003, Великий Новгород, ул. Б. Санкт-Петербургская, 41.

Отпечатано в ИПЦ НовГУ им. Ярослава Мудрого.
173003, Великий Новгород, ул. Б. Санкт-Петербургская, 41.