МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ

УЧРЕЖДЕНИЕ ОБРАЗОВАНИЯ
«БРЕСТСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»

Кафедра информатики и прикладной математики

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по выполнению контрольной работы
по дисциплине **«Дискретная математика»**
для студентов специальности
1 – 36 01 01 «Технология машиностроения»
заочной формы обучения

БРЕСТ 2013

УДК 510.22

Методические указания содержат формулировки заданий, краткие теоретические сведения и примеры выполнения заданий контрольной работы по дисциплине «**Дискретная математика**». В настоящее издание включены следующие три задачи: задача о наилучшей загрузке станка, задача составления расписания горячей обработки и задача о максимальном потоке в сетевой модели технологического процесса.

Предназначены для студентов второго курса специальности «*Технология машиностроения*» специализации «*Технология автоматизированного производства*» заочной формы обучения.

Составители: Тузик И.В. ст. преподаватель
Хомицкая Т.Г., ст. преподаватель
Рамская Л.К., ст. преподаватель

Требования к выполнению и оформлению контрольной работы.

Данная контрольная работа состоит из трех заданий. Вариант, по которому выполняется контрольная работа, выбирается по двум последним цифрам номера зачетной книжки или назначается преподавателем. Контрольная работа выполняется студентом самостоятельно и оформляется на листах формата А4.

Для каждого задания оформление выполняется по образцу:

№ задания. Постановка задачи (переписывается из данного методического пособия с указанием конкретных данных своего варианта).

Решение (записывается в произвольной форме, но аккуратно, логически последовательно и достаточно подробно; допускается построение расчетных таблиц в табличном процессоре MS Excel, в этом случае к решению прилагаются распечатки).

Ответ (записывается отдельно или выделяется в тексте решения).

Для тех пунктов заданий, где необходимо использование компьютера (задание 1, п.3, задание 2, п.1 и п.3, задание 3, п.4), требуется приложить распечатки с числовыми результатами и формулами, которые использовались в ходе решения.

В заданиях 1 и 2 построение дерева решений обязательно!

Оформленная контрольная работа отсылается на проверку не позже, чем за две недели до начала сессии. Работа, выполненная не по своему варианту, возвращается студенту без проверки. Если в выполненной контрольной работе преподаватель обнаруживает ошибки, то контрольная работа возвращается студенту для их исправления. Работа над ошибками прилагается к ранее выполненной работе. Не удаляйте листы с ошибками, а также записи с замечаниями рецензента! Правильно выполненная контрольная работа получает отметку «допуск», остается у преподавателя и является одним из условий допуска студента к зачету по соответствующей дисциплине.

Задание 1. **Задача о наилучшей загрузке станка**

Постановка задачи. Станок налажен на обработку деталей определенной группы. Выделен фонд времени непрерывной работы станка с данной наладкой. Известно число деталей в группе и длительности обработки деталей, а также ценность каждой детали в группе. Предполагается, что вся группа не может быть обработана в течение выделенного фонда времени, длительности обработки деталей могут быть разными, а очередность обработки деталей не важна. Требуется отобрать из группы детали для загрузки станка в течение выделенного фонда времени так, чтобы суммарная ценность отобранных деталей была максимальной.

Задания для выполнения.

1. *Составить и записать математическую модель поставленной задачи в соответствии с данными своего варианта.*
2. Найти **все** решения данной задачи методом ветвей и границ.
3. Решить поставленную задачу в Excel с помощью надстройки «Поиск решения», сформировав *Отчет по результатам*, распечатку которого также приложить к работе.

Задание 2. **Составление расписания горячей обработки**

Постановка задачи. На линии горячей обработки, состоящей из 4-х станков M_1, M_2, M_3, M_4, нужно обработать партию деталей D_1, D_2, D_3, D_4, D_5, D_6. Все детали должны проходить по линии в одном направлении через каждый станок. Порядок, в котором обрабатываются детали, может быть произвольным. Длительности p_{sf} обработки деталей на станках заданы матрицей P (см. свой вариант). Горячая обработка отличается тем свойством, что каждая деталь может ожидать обработку только перед первым станком линии, а перед остальными – ожидание недопустимо. Требуется составить расписание горячей обработки данной партии деталей за минимальное время.

Задания для выполнения.

1. По заданной матрице P получить матрицу W размерности 7x7, где w_{ij} - время между моментами начала обработки 1-м станком деталей D_i и D_j по любому компактному расписанию, где детали следуют одна за другой в порядке D_i, D_j. 7-я деталь – фиктивная.
2. Для полученной матрицы W найти оптимальный порядок обработки деталей за минимальное время. Построить диаграмму, соответствующую найденному порядку.
3. Решить поставленную задачу в Excel с помощью надстройки «Поиск решения», сформировав *Отчет по результатам*, распечатку которого также приложить к работе.

Задание 3. **Сетевая модель технологического процесса**

Постановка задачи. Для достижения заданных чертежом размеров и технических требований все поверхности детали проходят несколько стадий обработки, преобразующих ее из состояния заготовки в состояние готовой поверхности. Предполагается, что поверхность заготовки можно обработать не единственным способом (например, разными видами оборудования). Требуется так распределить заготовки по доступному оборудованию, чтобы число заготовок, обрабатываемых в единицу времени, было максимальным.

В этом случае задачу можно представить в виде сети. Вершины сети – это различные состояния поверхности, начиная от заготовки (состояние 1) и заканчивая готовой поверхностью (состояние 8). Источник – это заготовка в исходном (необработанном) состоянии, а сток – это готовая поверхность. Дуга сети обозначает оборудование, с помощью которого поверхность переходит из одного промежуточного состояния в другое. Для каждого доступного вида оборудования, которое переводит заготовку из состояния i в состояние j, известна величина c_{ij}. c_{ij} - это число заготовок, которое может быть обработано в единицу времени (т.е. пропускная способность соответствующей дуги).

Сетевая модель технологического процесса обработки представлена матрицей С. На пересечении строки i и столбца j указано максимально допустимое количество заготовок, которое может быть обработано на данном оборудовании в единицу времени при переходе заготовки из состояния i в состояние j.

Задания для выполнения.

1. По заданной матрице С построить сеть, соответствующую условию задачи.
2. Найти любой полный поток в построенной сети и указать его величину.
3. По полученному полному потоку найти максимальный поток и указать его величину.
4. Решить поставленную задачу в Excel с помощью надстройки «Поиск решения», сформировав *Отчет по результатам*, его распечатку также приложить к работе.
5. Указать оборудование, которое оказалось загружено наибольшим количеством заготовок а) в решении, полученном вручную, б) в решении, полученном в Excel.

1.1. Пример решения задачи о наилучшей загрузке станка методом ветвей и границ

Задача. Станок налажен на обработку деталей определенной группы. Выделен фонд времени непрерывной работы станка с данной наладкой. Известно число деталей в группе, длительности их обработки и ценность каждой детали в группе. Предполагается, что вся группа не может быть обработана в течение выделенного фонда времени, длительности обработки деталей могут быть разными, а очередность обработки деталей не важна. Требуется отобрать из группы детали для загрузки станка в течение выделенного фонда времени так, чтобы общая ценность отобранных деталей была максимальной.

Пример.

$$3\cdot x_1 + 6\cdot x_2 + 5\cdot x_3 + 3\cdot x_4 + 2\cdot x_5 + 3\cdot x_6 \to \max$$

$$7\cdot x_1 + 10\cdot x_2 + 8\cdot x_3 + 4\cdot x_4 + 2\cdot x_5 + 5\cdot x_6 \le 23,\ x_i \in \{0,1\},\ i=1,\dots,6.$$

Найдем удельную ценность каждой детали (ценность на единицу времени). В последнем столбце в таблице слева показан порядок расположения деталей в группе по убыванию их удельных ценностей.

i	c_i	t_i	c_i/t_i	Порядок
1	3	7	3/7	⑥
2	6	10	6/10	④
3	5	8	5/8	③
4	3	4	3/4	②
5	2	2	1	①
6	3	5	3/5	⑤

Расположим детали в порядке убывания их удельных ценностей (см. таблицу справа). *Примечание.* Если несколько деталей имеет одинаковую удельную ценность (в примере – 2-я и 6-я), то они считаются равноценными, и между собой могут быть расположены в произвольном порядке.

i	c_i	t_i	c_i/t_i	
5	2	2	1	①
4	3	4	3/4	②
3	5	8	5/8	③
2	6	10	3/5	④
6	3	5	3/5	⑤
1	3	7	3/7	⑥

Учитывая этот порядок, найдем верхнюю оценку общей ценности выбранных деталей (ценность бóльшая, чем эта оценка, уже не может быть получена в процессе решения задачи). Последовательно выбираем детали в указанном порядке, пока общее время об-работки при выборе очередной детали не превысит заданного (23). В нашем случае 5-я, 4-я, 3-я детали выбираются целиком, поэтому $x_5=x_4=x_3=1$. Их обработка займет $2\cdot x_5+4\cdot x_4+8\cdot x_3=14$ единиц времени, а следующая, 2-я деталь, требует для своей обработки 10 единиц времени и не может быть целиком обработана за оставшиеся 23-14=9 единиц времени. Поэтому берем только часть 2-й детали такую, которая может быть обработана в точности за оставшиеся 9 единиц времени, т.е. $x_2 = 9/10$. При этом последние в списке, 6-я и 1-я детали, вообще не могут быть выбраны, т.к. выделенный фонд времени исчерпан, поэтому $x_6=x_1=0$. Тогда общее время обработки выбранных деталей составит $2\cdot x_5+4\cdot x_4+8\cdot x_3+10\cdot x_2+5\cdot x_6+7\cdot x_1=2\cdot 1+4\cdot 1+8\cdot 1+10\cdot 9/10+5\cdot 0+7\cdot 0=23$, а их суммарная ценность – $2\cdot x_5+3\cdot x_4+5\cdot x_3+6\cdot x_2+3\cdot x_6+3\cdot x_1= 2\cdot 1+3\cdot 1+5\cdot 1+6\cdot 9/10+3\cdot 0+3\cdot 0=15\ 2/5$. Т.е. верхняя оценка общей ценности равна 15 2/5. Эта величина является именно оценкой, т.к. при ее вычислении использовались дробные значения x_i. В процессе решения задачи эта величина может только уменьшиться (в лучшем случае, суммарная ценность окажется равной 15). Все полученные в данном рассуждении результаты сведены в таблицу №0:

0)

i	c_i	t_i	x_i	$\Sigma(t_i\cdot x_i)$	$\Sigma(c_i\cdot x_i)$
5	2	2	1	2	2
4	3	4	1	6	5
3	5	8	1	14	10
2	6	10	9/10	23	15 2/5
6	3	5	0	23	15 2/5
1	3	7	0	23	15 2/5

Здесь $\Sigma(t_i \cdot x_i)$ – суммарное время обработки деталей, а $\Sigma(c_i \cdot x_i)$ – их суммарная ценность.

Начнем процесс применения метода ветвей и границ. Ветвление будем вести по тем деталям, для которых в процессе решения будут получаться дробные значения x_i. Например, в предыдущей таблице дробное значение было получено для детали с номером 2 ($x_2 = 9/10$). Поэтому далее будем рассматривать два случая: $x_2 = 1$ или $x_2 = 0$, т.е. либо брать в обработку 2-ю деталь, либо нет.

Процесс решения будем сопровождать построением дерева. В корневой вершине дерева указываем верхнюю оценку общей ценности, полученную в таблице с номером 0 (у нас это 15 2/5). На каждой последующей вершине дерева будем указывать значение выбранного x_i (x_i=0 или x_i=1) и соответствующую оценку суммарной ценности (или ее точное значение при целочисленных x_i , i=1,...,6). Ту из открытых вершин дерева, где оценка общей ценности окажется больше, выбираем для дальнейшего ветвления.

Из вершины дерева НЕ строим ветви, если для нее выполняется одно из условий:

– в этой вершине нет решений (превышено выделенное время обработки деталей);
– в этой вершине найдено значение целевой функции при целочисленных значениях x_i , i=1,...,6;
– в этой вершине полученная оценка меньше, чем уже найденное значение целевой функции при целочисленных значениях x_i , i=1,...,6.

Построение таблиц

В таблице 0 дробное значение принимает $x_2 = 9/10$. Значит, выполняем ветвление по детали с номером 2. Этому ветвлению будет соответствовать новая таблица №1. Ее получаем следующим образом:

– возьмем данные (столбцы i, c_i, t_i) из предыдущей таблицы с номером 0;
– строку, соответствующую 2-й детали, запишем в качестве первой строки новой таблицы;
– остальные строки таблицы 0 сместим на одну вниз, оставив их порядок без изменения.
– добавляем столбцы x_i, $\Sigma(t_i \cdot x_i)$, $\Sigma(c_i \cdot x_i)$ для $x_2 = 1$ и такие же столбцы для $x_2 = 0$;
– для $x_2 = 1$ заполняем сначала столбцы x_i и $\Sigma(t_i \cdot x_i)$:
  полагаем $x_2 = 1$ (это обязательное условие), тогда $\Sigma(t_i \cdot x_i)=10\cdot1=10$, из 23 остаются свободными 13 единиц;
  деталь 5 требует 2 единицы времени для обработки, на данный момент имеется 13 свободных единиц времени, значит, полагаем $x_5 = 1$, тогда $\Sigma(t_i \cdot x_i)=10\cdot1+2\cdot1=12$, из 23 остаются свободными 11 единиц;
  деталь 4 требует 4 единицы времени для обработки, на данный момент имеется 11 свободных единиц времени, значит, полагаем $x_4 = 1$, тогда $\Sigma(t_i \cdot x_i)=10\cdot1+2\cdot1+4\cdot1=16$, из 23 остаются свободными 7 единиц;
  деталь 3 требует 8 единиц времени для обработки, на данный момент имеется только 7 свободных единиц времени, тогда $x_3 = 7/8$, $\Sigma(t_i \cdot x_i)=10\cdot1+2\cdot1+4\cdot1+8\cdot7/8=23$, из 23 не остается свободных единиц времени. Значит, $x_6 = 0$ и $x_1 = 0$.
– По найденным x_i заполняем столбец $\Sigma(c_i \cdot x_i)$ и получаем оценку суммарной ценности взятых деталей при $x_2 = 1$: $\Sigma(ci \cdot xi) = 6\cdot1+2\cdot1+3\cdot1+5\cdot7/8+3\cdot0+3\cdot0=15\ 3/8$.

– для $x_2 = 0$ заполняем сначала столбцы x_i и $\Sigma(t_i \cdot x_i)$:

полагаем $x_2 = 0$ (это обязательное условие), тогда $\Sigma(t_i \cdot x_i)=10 \cdot 0=0$, из 23 остаются свободными 23 единицы;

деталь 5 требует 2 единицы времени для обработки, на данный момент имеется 23 свободных единиц времени, значит, полагаем $x_5 = 1$, тогда $\Sigma(t_i \cdot x_i)=10 \cdot 0+2 \cdot 1=2$, из 23 остается свободной 21 единица;

деталь 4 требует 4 единицы времени для обработки, на данный момент имеется 21 свободная единица времени, значит, полагаем $x_4 = 1$, тогда $\Sigma(t_i \cdot x_i)=10 \cdot 0+2 \cdot 1+4 \cdot 1=6$, из 23 остаются свободными 17 единиц;

деталь 3 требует 8 единиц времени для обработки, на данный момент имеется 17 свободных единиц времени; полагаем $x_3=1$, тогда $\Sigma(t_i \cdot x_i)=10 \cdot 0+2 \cdot 1+4 \cdot 1+8 \cdot 1=14$, из 23 остаются свободными 9 единиц;

деталь 6 требует 5 единиц времени для обработки, на данный момент имеется 9 свободных единиц времени; полагаем $x_6 = 1$, тогда $\Sigma(t_i \cdot x_i)=10 \cdot 0+2 \cdot 1+4 \cdot 1+8 \cdot 1+5 \cdot 1=19$, из 23 остаются свободными 4 единицы;

деталь 1 требует 7 единиц времени для обработки, на данный момент имеется 4 свободных единицы времени, тогда $x_1=4/7$, $\Sigma(t_i \cdot x_i)=10 \cdot 0+2 \cdot 1+4 \cdot 1+8 \cdot 1+5 \cdot 1+7 \cdot 4/7=23$, из 23 не остается свободных единиц времени.

– По найденным x_i заполняем столбец $\Sigma(c_i \cdot x_i)$ и получаем оценку суммарной ценности взятых деталей при $x_2 = 0$: $\Sigma(c_i \cdot x_i) = 6 \cdot 0+2 \cdot 1+3 \cdot 1+5 \cdot 1+3 \cdot 1+3 \cdot 4/7=14\ 5/7$.

Строим соответствующее ветвление на дереве, а затем из всех открытых в данный момент вершин выбираем ту, у которой оценка больше (это вершина с оценкой 15 3/8).

Значит, далее будем рассматривать таблицу 1 при $x_2 = 1$ (т.е. таблицу, где получена оценка, соответствующая выбранной вершине). В этой таблице $x_3 = 7/8$ принимает дробное значение, значит, ветвление будем вести по 3-ей детали. Этому ветвлению будет соответствовать новая таблица (с номером 2). Ее строим аналогично предыдущей таблице, но для нее нужно учитывать новое обязательное условие: $x_2 = 1$.

Остальные таблицы строятся с помощью аналогичных рассуждений. Все таблицы, которые пришлось построить при решении данной задачи, приведены ниже.

В верхней части каждой таблицы (начиная с таблицы с номером 2) указываются обязательные условия, при которых ведутся расчеты в таблице. В качестве этих условий берутся значения x_i, расположенных выше по дереву по отношению к рассматриваемой вершине (*по которой ведется ветвление*).

Расчетные таблицы могут быть построены вручную, либо с использованием MS Excel.

При построении в Excel можно создать одну расчетную таблицу так, как показано на экранной копии. При проведении расчетов в такой таблице придется менять вручную только значения столбца " i " и двух столбцов " x_i ". После заполнения этой таблицы можно ее копировать и табличные значения, рассчитанные по формулам, размещать на рабочем листе, используя контекстное меню, *Специальная вставка...*, ⊙*значения*.

Вид заполненной таблицы:

	A	B	C	D	E	F	G	H	I	J
1	23	1	2	3	4	5	6			
2	ti	3	6	5	3	2	3			
3	ci	7	10	8	4	2	5			
4										
5		i	ci	ti	xi	Σti·x	Σci·x	xi	Σti·x	Σci·xi
6		1	3	7	1	7	3	0	0	0
7		3	5	8	0	7	3	0	0	0
8		2	6	10	1	17	9	1	10	6
9		5	2	2	1	19	11	1	12	8
10		4	3	4	1	23	14	1	16	11
11		6	3	5	0	23	14	1	21	14

Та же таблица в формульном виде:

	A	B	C	D	E	F	G	H	I	J
1	23	1	2	3	4	5	6			
2	ti	3	6	5	3	2	3			
3	ci	7	10	8	4	2	5			
4										
5		i	ci	ti	xi	Σti·xi	Σci·xi	xi	Σti·xi	Σci·xi
6		1	=ПРОСМОТР(B6;$B$1:$G$1;$B$2:$G$2)	=ПРОСМОТР(B6;$B$1:$G$1;$B$3:$G$3)	1	=$D$6*E6	=$C$6*E6	0	=$D$6*H6	=$C$6*H6
7		3	=ПРОСМОТР(B7;$B$1:$G$1;$B$2:$G$2)	=ПРОСМОТР(B7;$B$1:$G$1;$B$3:$G$3)	0	=F6+$D$7*E7	=G6+$C$7*E7	0	=I6+$D$7*H7	=J6+$C$7*H7
8		2	=ПРОСМОТР(B8;$B$1:$G$1;$B$2:$G$2)	=ПРОСМОТР(B8;$B$1:$G$1;$B$3:$G$3)	1	=F7+$D$8*E8	=G7+$C$8*E8	1	=I7+$D$8*H8	=J7+$C$8*H8
9		5	=ПРОСМОТР(B9;$B$1:$G$1;$B$2:$G$2)	=ПРОСМОТР(B9;$B$1:$G$1;$B$3:$G$3)	1	=F8+$D$9*E9	=G8+$C$9*E9	1	=I8+$D$9*H9	=J8+$C$9*H9
10		4	=ПРОСМОТР(B10;$B$1:$G$1;$B$2:$G$2	=ПРОСМОТР(B10;$B$1:$G$1;$B$3:$G$3	1	=F9+$D$10*E10	=G9+$C$10*E10	1	=I9+$D$10*H10	=J9+$C$10*H10
11		6	=ПРОСМОТР(B11;$B$1:$G$1;$B$2:$G$2	=ПРОСМОТР(B11;$B$1:$G$1;$B$3:$G$3	0	=F10+$D$11*E11	=G10+$C$11*E11	1	=I10+$D$11*H11	=J10+$C$11*H11
12										

1.2. Пример оформления решения задачи о наилучшей загрузке станка

Математическая модель:

$$3\cdot x_1+6\cdot x_2+5\cdot x_3+3\cdot x_4+2\cdot x_5+3\cdot x_6 \to \max$$

$$7\cdot x_1+10\cdot x_2+8\cdot x_3+4\cdot x_4+2\cdot x_5+5\cdot x_6 \le 23$$

$$x_i \in \{0,1\},\ i=1,\dots,6.$$

i	c_i	t_i	c_i/t_i	Порядок
1	3	7	3/7	⑥
2	6	10	6/10	④
3	5	8	5/8	③
4	3	4	3/4	②
5	2	2	1	①
6	3	5	3/5	⑤

0)	i	c_i	t_i	x_i	$\Sigma(t_i\cdot x_i)$	$\Sigma(c_i\cdot x_i)$
	5	2	2	1	2	2
	4	3	4	1	6	5
	3	5	8	1	14	10
	2	6	10	9/10	23	15 2/5
	6	3	5	0	23	15 2/5
	1	3	7	0	23	15 2/5

1)								
из т.0)			$x_2=1$			$x_2=0$		
i	c_i	t_i	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$
2	6	10	**1**	10	6	**0**	0	0
5	2	2	1	12	8	1	2	2
4	3	4	1	16	11	1	6	5
3	5	8	7/8	23	15 3/8	1	14	10
6	3	5	0	23	15 3/8	1	19	13
1	3	7	0	23	15 3/8	4/7	23	14 5/7

2)			$x_2=1$					
из т.1)			$x_3=1$			$x_3=0$		
i	c_i	t_i	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$
3	5	8	**1**	8	5	**0**	0	0
2	6	10	**1**	18	11	**1**	10	6
5	2	2	1	20	13	1	12	8
4	3	4	3/4	23	15 1/4	1	16	11
6	3	5	0	23	15 1/4	1	21	14
1	3	7	0	23	15 1/4	2/7	23	14 6/7

3)			$x_2=1,\ x_3=1$					
из т.2)			$x_4=1$			$x_4=0$		
i	c_i	t_i	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$
4	3	4	**1**	4	3	**0**	0	0
3	5	8	**1**	12	8	1	8	5
2	6	10	**1**	22	14	1	18	11
5	2	2	1/2	23	15	1	20	13
6	3	5	0	23	15	3/5	23	14 4/5
1	3	7	0	23	15	0	23	14 4/5

4)			$x_2=1,\ x_3=1,\ x_4=1$					
из т.3)			$x_5=1$			$x_5=0$		
i	c_i	t_i	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$
5	2	2	1	2		**0**	0	0
4	3	4	1	6		**1**	4	3
3	5	8	1	14		**1**	12	8
2	6	10	1	24		**1**	22	14
6	3	5	☹ нет решений			1/5	23	14 3/5
1	3	7				0	23	14 3/5

5)			$x_2=1, x_3=0$					
из т.2)			$x_1=1$			$x_1=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
1	3	7	**1**	7	3	**0**	0	0
3	5	8	**0**	7	3	**0**	0	0
2	6	10	**1**	17	9	**1**	10	6
5	2	2	1	19	11	1	12	8
4	3	4	1	23	14	1	16	11
6	3	5	0	23	14☺	1	21	14☺

6)			$x_2=1, x_3=1, x_4=0$					
из т.3)			$x_6=1$			$x_6=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
6	3	5	**1**	5	3	**0**	0	0
4	3	4	**0**	5	3	**0**	0	0
3	5	8	**1**	13	8	1	8	5
2	6	10	**1**	23	14	1	18	11
5	2	2	0	23	14	1	20	13
1	3	7	0	23	14☺	3/7	23	14 2/7

7)			$x_2=0$					
из т.1)			$x_1=1$			$x_1=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
1	3	7	**1**	7	3	**0**	0	0
2	6	10	**0**	7	3	**0**	0	0
5	2	2	1	9	5	1	2	2
4	3	4	1	13	8	1	6	5
3	5	8	1	21	13	1	14	10
6	3	5	2/5	23	14 1/5	1	19	13

8)			$x_2=1, x_3=1, x_4=1, x_5=0$					
из т.4)			$x_6=1$			$x_6=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
6	3	5	**1**	5		**0**	0	0
5	2	2	**0**	5		**0**	0	0
4	3	4	**1**	9		1	4	3
3	5	8	**1**	17		1	12	8
2	6	10	**1**	27	☹	1	22	14
1	3	7	нет решений			1/7	23	14 3/7

9)			$x_2=1, x_3=1, x_4=1, x_5=0, x_6=0$					
из т.8)			$x_1=1$			$x_1=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
1	3	7	**1**	7		**0**	0	0
6	3	5	**0**	7		**0**	0	0
5	2	2	**0**	7		**0**	0	0
4	3	4	**1**	11		**1**	4	3
3	5	8	**1**	19	☹	**1**	12	8
2	6	10	**1**	29	н/р	**1**	22	14☺

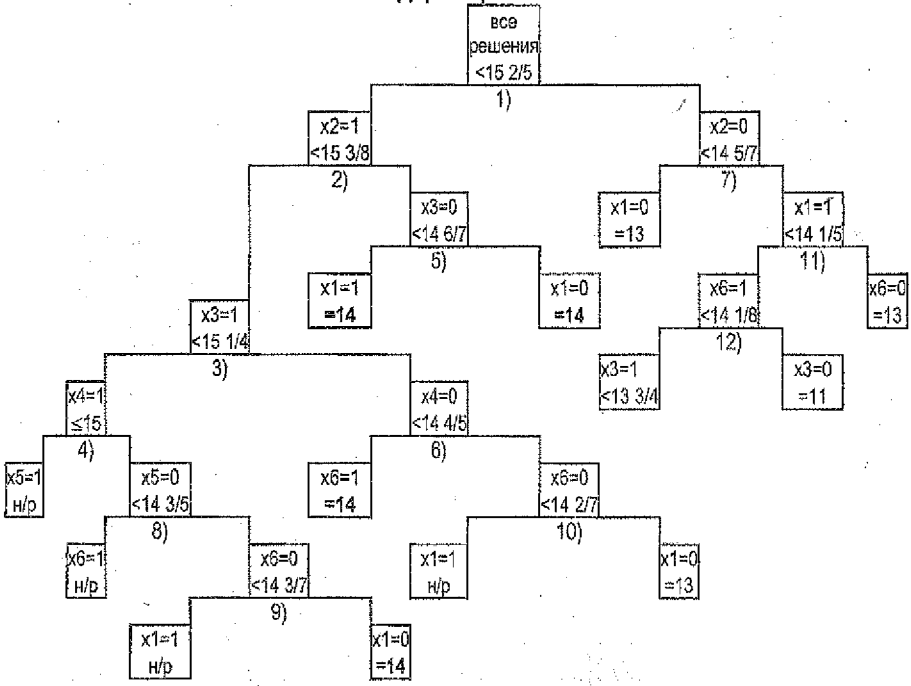
10)			$x_2=1, x_3=1, x_4=0, x_6=0$					
из т.6)			$x_1=1$			$x_1=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
1	3	7	**1**	7		**0**	0	0
6	3	5	**0**	7		**0**	0	0
4	3	4	**0**	7		**0**	0	0
3	5	8	**1**	15		**1**	8	5
2	6	10	**1**	25	☹	**1**	18	11
5	2	2	нет решений			1	20	13

11)			$x_2=0, x_1=1$					
из т.7)			$x_6=1$			$x_6=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
6	3	5	**1**	5	3	**0**	0	0
1	3	7	**1**	12	6	**1**	7	3
2	6	10	**0**	12	6	**0**	7	3
5	2	2	1	14	8	1	9	5
4	3	4	1	18	11	1	13	8
3	5	8	5/8	23	14 1/8	1	21	13

12)			$x_2=0, x_1=1, x_6=1$					
из т.11)			$x_3=1$			$x_3=0$		
i	c_i	t_i	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$	x_i	$\Sigma t_i \cdot x_i$	$\Sigma c_i \cdot x_i$
3	5	8	1	8	5	**0**	0	0
6	3	5	1	13	8	**1**	5	3
1	3	7	1	20	11	**1**	12	6
2	6	10	**0**	20	11	**0**	12	6
5	2	2	1	22	13	1	14	8
4	3	4	1/4	23	13 3/4	1	18	11

Ответ: 1-е решение - $(x_1, x_2, x_3, x_4, x_5, x_6)=(1,1,0,1,1,0)$, общее время обработки равно 23;
2-е решение - $(x_1, x_2, x_3, x_4, x_5, x_6)=(0,1,0,1,1,1)$, общее время обработки равно 21;
3-е решение - $(x_1, x_2, x_3, x_4, x_5, x_6)=(0,1,1,0,0,1)$, общее время обработки равно 23;
4-е решение - $(x_1, x_2, x_3, x_4, x_5, x_6)=(0,1,1,1,0,0)$, общее время обработки равно 22.
Максимальная общая ценность выбранных для обработки деталей равна 14.

Дерево решений:



Все вершины, на которых можно было получить максимальную ценность, рассмотрены.

Примечание. Поясним построение дерева решений. Ветвления соответствуют таблицам. На 1-м ветвлении ($x_2=1$, $x_2=0$) выбираем вершину, где $x_2=1$, т.к. при этом условии есть шансы получить ценность, равную 15, а при $x_2=0$ – не более 14.

После выполнения 2-го ветвления открытыми остаются вершины с оценками общей ценности $15\,^1/_4$ ($x_3=1$), $14\,^6/_7$ ($x_3=0$), $14\,^5/_7$ ($x_2=0$). Из них выбираем вершину с оценкой $15\,^1/_4$ ($x_3=1$), где остаются шансы получить ценность, равную 15.

После выполнения 3-го ветвления открытыми остаются вершины с оценками общей ценности 15 ($x_4=1$), $14\,^4/_5$ ($x_4=0$), $14\,^6/_7$ ($x_3=0$), $14\,^5/_7$ ($x_2=0$). Из них выбираем вершину с оценкой 15 ($x_4=1$), где остаются шансы получить ценность, равную 15.

После выполнения 4-го ветвления видим, что при $x_5=1$, с учетом того, что было выше по дереву ($x_4=x_3=x_2=1$), время обработки выбранных деталей превышает заданное, значит, невозможно одновременное выполнение всех этих условий, поэтому дальнейшее ветвление из этой вершины не производим (нет решений).

Далее из всех открытых на данный момент вершин выбираем ту, где оценка ценности наибольшая – $14\,^6/_7$ ($x_3=0$). Ценность, большую, чем 14, мы уже нигде не получим.

На 5-м ветвлении получаем сразу 2 решения – это наборы с ценностью, равной 14. Это уже не оценка, а точное значение ценности (т.к. оно получено при целочисленных x_i).

Из вершин, где получено точное значение ценности, а так же из тех вершин, где оценка ценности меньше, чем уже найденное точное значение($13\,^3/_4$), ветвлений не делаем.

1.3. Пример решения задачи о наилучшей загрузке станка в Excel

Математическая модель:

$$3 \cdot x_1 + 6 \cdot x_2 + 5 \cdot x_3 + 3 \cdot x_4 + 2 \cdot x_5 + 3 \cdot x_6 \to \max$$

$$7 \cdot x_1 + 10 \cdot x_2 + 8 \cdot x_3 + 4 \cdot x_4 + 2 \cdot x_5 + 5 \cdot x_6 \le 23$$

$$x_i \in \{0,1\},\ i=1,\ldots,6.$$

Найдем максимальную ценность деталей при заданных условиях, используя надстройку *«Поиск решения»* (при этом будет найден только один из наборов (x_1, x_2, x_3, x_4, x_5, x_6)).

- Подготовим исходные данные на рабочем листе Excel.

а) Введем числовые данные (диапазон для неизвестных x_i оставляем пустым):

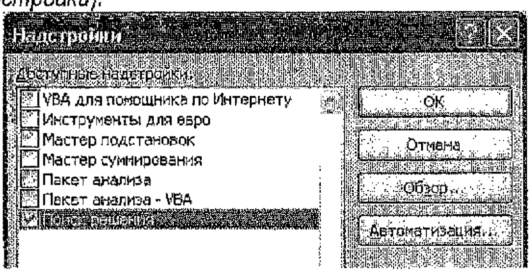
	A	B	C	D	E	F	G	H
1	i	1	2	3	4	5	6	
2	ci	3	6	5	3	2	3	
3	ti	7	10	8	4	2	5	
4	xi							

б) Введем формулы для $\Sigma(c_i \cdot x_i)$ и $\Sigma(t_i \cdot x_i)$:

	A	B	C	D	E	F	G	H
1	i	1	2	3	4	5	6	
2	ci	3	6	5	3	2	3	=СУММПРОИЗВ(B2:G2;$B$4:$G$4)
3	ti	7	10	8	4	2	5	=СУММПРОИЗВ(B3:G3;$B$4:$G$4)
4	xi							

- Подготовим исходные данные в окне *«Поиск решения»*.

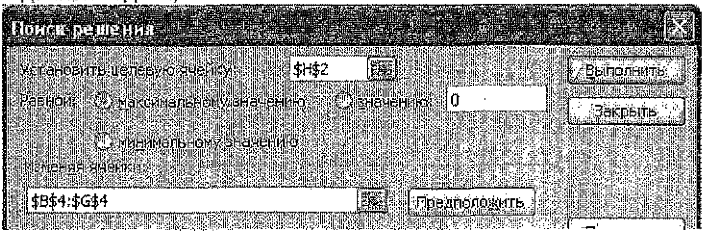
а) В меню *Сервис* выберем команду *«Поиск решения»*. Если такая команда в меню *Сервис* отсутствует, то предварительно подключим соответствующую надстройку (меню *Сервис*, *Надстройки*):



б) В окне *«Поиск решения»*:

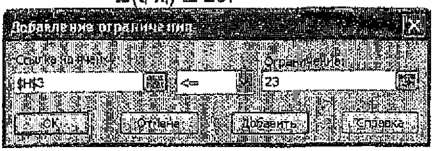
- укажем *целевую ячейку* с формулой, задающей целевую функцию, чье максимальное значение требуется найти (у нас это ячейка H2, содержащая формулу $\Sigma(c_i \cdot x_i)$);
- укажем, что требуется найти именно *максимальное значение*;
- укажем *изменяемые ячейки* с неизвестными значениями x_i, $i=1,\ldots,6$ (у нас это диапазон ячеек B4:G4); указав эти ячейки как изменяемые, мы разрешаем Excel

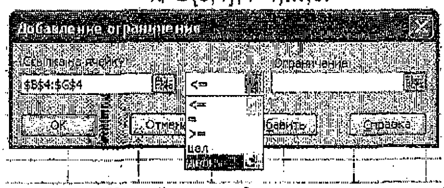
изменять значения в этих ячейках, подбирая наиболее подходящие для решения данной задачи;



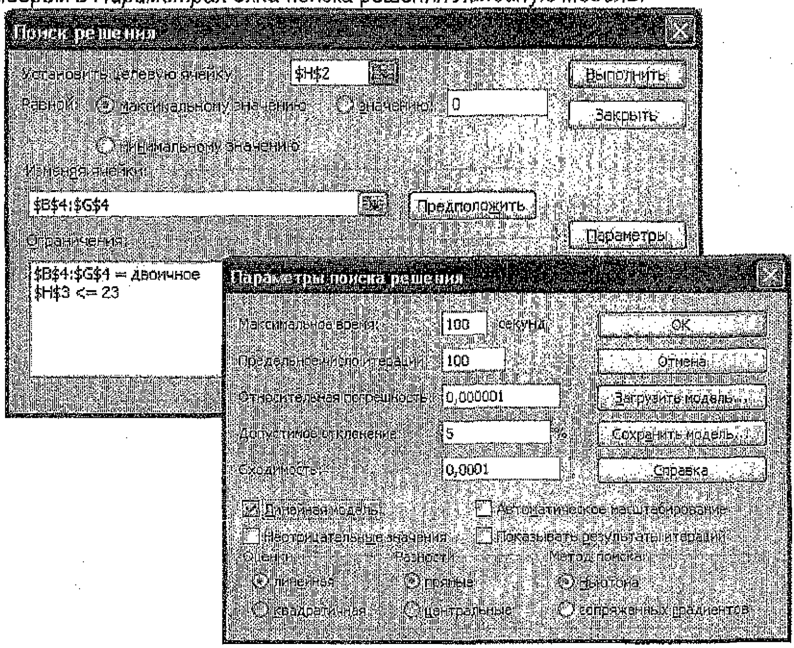
- добавим ограничения из математической модели задачи:

$\Sigma(t_i \cdot x_i) \leq 23$; $x_i \in \{0,1\}$, $i=1,\ldots,6$;

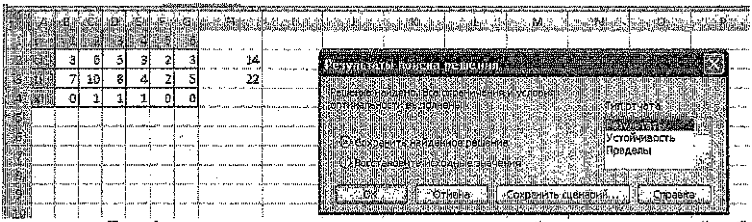




- выберем в *Параметрах* окна поиска решения *Линейную модель*:



- Получим решение, щелкнув кнопку *Выполнить* в окне *«Поиск решения»*:



Видим, что Excel нашел одно из возможных решений задачи: для обработки выбираем 2-ю, 3-ю и 4-ю детали, т.к. $(x_1,x_2,x_3,x_4,x_5,x_6)=(0,\ 1,\ 1,\ 1,\ 0,\ 0)$, с максимальной суммарной ценностью, равной 14 (такая же ценность была получена при решении задачи вручную). При этом все выбранные детали могут быть обработаны за время, равное 22 (не превышающее заданного).

- Создадим *Отчет по результатам* (он будет выведен на отдельном листе рабочей книги), выбрав в появившемся окне «Результаты поиска решения» *Тип отчета - Результаты*:

Microsoft Excel 12.0 Отчет по результатам
Рабочий лист: [перебор_new.xlsm]ЛР№4
Отчет создан: 08.11.2013 22:01:01

Целевая ячейка (Максимум)

Ячейка	Имя	Исходное значение	Результат
$H$2	ci	0	14

Изменяемые ячейки

Ячейка	Имя	Исходное значение	Результат
$B$4	xi	0	0
$C$4	xi	0	1
$D$4	xi	0	1
$E$4	xi	0	1
$F$4	xi	0	0
$G$4	xi	0	0

Ограничения

Ячейка	Имя	Значение	Формула	Статус	Разница
$H$3	fi	22	$H$3<=23	не связан.	1
$B$4	xi	0	$B$4=двоичное	связанное	0
$C$4	xi	1	$C$4=двоичное	связанное	0
$D$4	xi	1	$D$4=двоичное	связанное	0
$E$4	xi	1	$E$4=двоичное	связанное	0
$F$4	xi	0	$F$4=двоичное	связанное	0
$G$4	xi	0	$G$4=двоичное	связанное	0

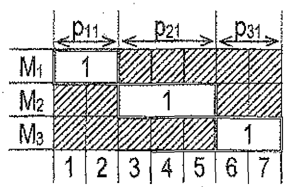
Лист1 Отчет по результатам 1

2.1. Составление расписания горячей обработки (РГО)

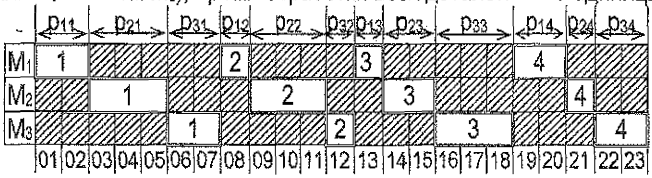
Задача. На линии горячей обработки, состоящей из нескольких станков, нужно обработать партию деталей. Все детали должны проходить по линии в одном направлении через каждый станок. Заданы длительности обработки деталей каждым станком. Горячая обработка отличается тем свойством, что каждая деталь может ожидать обработку только перед первым станком линии, а перед остальными – ожидание недопустимо. Иными словами, войдя в линию, каждая деталь должна обрабатываться непрерывно – одним станком за другим. Требуется составить расписание горячей обработки данной партии деталей за минимальное время, т.е. минимальное по длине расписание.

Пример расписания горячей обработки одной детали D_1 на линии с тремя станками M_1, M_2, M_3 (здесь матрицей P заданы длительности обработки деталей D_1, D_2, D_3, D_4 на каждом из станков):

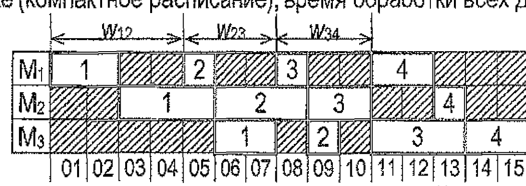
$$P = \begin{bmatrix} 2 & 1 & 1 & 2 \\ 3 & 3 & 2 & 1 \\ 2 & 1 & 3 & 2 \end{bmatrix}$$



Пример расписания горячей обработки четырех деталей D_1, D_2, D_3, D_4 на линии с тремя станками M_1, M_2, M_3 в порядке $D_1 \to D_2 \to D_3 \to D_4$ (через p_{st} обозначена длительность обработки детали D_t станком M_s), время обработки всех деталей – 23 единицы:



Пример расписания обработки деталей D_1, D_2, D_3, D_4 в порядке $D_1 \to D_2 \to D_3 \to D_4$, имеющего наименьшую длину среди всех расписаний, по которым детали обрабатываются в указанном порядке (компактное расписание), время обработки всех деталей – 15 единиц:



Для 4-х деталей число всех компактных расписаний равно 4!, т.е. числу всех перестановок этих деталей. Минимальное по длине расписание совпадает с одним из компактных.

Сведем задачу составления РГО к задаче коммивояжера.

Введем в рассмотрение величины w_{ij}, которые равны длине интервала между моментами начала обработки первым станком (M_1) деталей D_i и D_j по любому компактному расписанию, в котором эти детали следуют одна за другой в порядке D_i, D_j. Величина w_{ij} не зависит от временных характеристик деталей D_t при $t \neq i$, $t \neq j$, а зависит только от длительностей обработки деталей D_i, D_j. Примеры величин w_{ij} приведены выше на рисунке. Здесь, с учетом порядка обработки, представлены $w_{12}=4$, $w_{23}=3$, $w_{34}=3$.

Если линия состоит из k станков, а число обрабатываемых деталей равно m, то величины w_{ij} рассчитываются по формуле:

$$w_{ij} = \max_{1 \le h \le k}\left\{\sum_{s=1}^{h} p_{si} - \sum_{s=1}^{h-1} p_{sj}\right\}, i,j = 1,2,\ldots,m.$$

Полученные таким образом величины образуют матрицу размерности $m \times m$. Введем в рассмотрение фиктивную ($m+1$)-ю деталь, которая входит в линию последней и проходит обработку на всех станках мгновенно, за 0 единиц времени. Она нужна, чтобы учесть полное время обработки той из реальных деталей, которая обрабатывается последней. Добавим в нашу матрицу ($m+1$)-ю строку, состоящую из всех нулей ($w_{m+1,j}=0$, $j=1,2,\ldots,m$), и ($m+1$)-й столбец, в котором каждый элемент рассчитывается по формуле: $w_{i,m+1}=p_{1i} + p_{2i} +\ldots+ p_{ki}$, $i= 1,2,\ldots,m$, т.е. для каждой детали вычисляется ее общее время обработки на всех станках.

Пример получения матрицы величин w_{ij} по заданной матрице P в *Derive* (без последней строки и последнего столбца). Вводим матрицу P. Для этого в строке ввода в квадратных скобках перечисляем через запятую элементы матрицы P, отделяя строки друг от друга точкой с запятой:

p:=[1, 5, 7, 5, 3, 4; 1, 5, 4, 3, 2, 6; 5, 3, 4, 6, 1, 4; 2, 4, 1, 2, 3, 6]

Нажав клавишу Enter, получим:

$$\#1:\ p := \begin{bmatrix} 1 & 5 & 7 & 5 & 3 & 4 \\ 1 & 5 & 4 & 3 & 2 & 6 \\ 5 & 3 & 4 & 6 & 1 & 4 \\ 2 & 4 & 1 & 2 & 3 & 6 \end{bmatrix}$$

Далее вводим выражение для получения матрицы W (пока без 7-й строки и 7-го столбца):

vector(vector(max(vector(sum(p sub s sub i, s, 1, h) – sum(p sub s sub j, s, 1, h-1), h, 1, 4)), j, 1, 6), i, 1, 6)

Здесь 4 – число станков, а 6 – число деталей. Нажав Ctrl+Enter, получим:

$$\#2:\ \mathrm{VECTOR}\left(\mathrm{VECTOR}\left(\mathrm{MAX}\left(\mathrm{VECTOR}\left(\left(\sum_{s=1}^{h} p_{s,i}\right) - \sum_{s=1}^{h-1} p_{s,j}, h, 1, 4\right)\right), j, 1, 6\right), i, 1, 6\right)$$

$$\#3:\ \begin{bmatrix} 5 & 1 & 1 & 1 & 3 & 1 \\ 11 & 5 & 5 & 5 & 11 & 6 \\ 13 & 7 & 7 & 7 & 10 & 7 \\ 12 & 5 & 5 & 6 & 10 & 5 \\ 4 & 3 & 3 & 3 & 3 & 3 \\ 13 & 7 & 5 & 6 & 14 & 6 \end{bmatrix}$$

Запишем полученную матрицу, добавив недостающие последнюю строку и последний столбец, соответствующие фиктивной детали (в нашем примере – 7-ю строку и 7-й столбец), а элементы главной диагонали заменив знаком ∞ :

$$W_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array}\begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} \infty & 1 & 1 & 1 & 3 & 1 & 9 \\ 11 & \infty & 5 & 5 & 11 & 6 & 17 \\ 13 & 7 & \infty & 7 & 10 & 7 & 16 \\ 12 & 5 & 5 & \infty & 10 & 5 & 16 \\ 4 & 3 & 3 & 3 & \infty & 3 & 9 \\ 13 & 7 & 5 & 6 & 14 & \infty & 20 \\ 0 & 0 & 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array}$$

Здесь значения 7-го столбца вычисляются по матрице P:
1+1+5+2=9 – время обработки 1-й детали на всех станках;
5+5+3+4=17 – время обработки 2-й детали на всех станках;
7+4+4+1=16 – время обработки 3-й детали на всех станках;
5+3+6+2=16 – время обработки 4-й детали на всех станках;
3+2+1+3=9 – время обработки 5-й детали на всех станках;
4+6+4+6=20 – время обработки 6-й детали на всех станках.
Значения 7-й строки равны 0,т.к. фиктивная деталь проходит обработку на всех станках мгновенно, за время, равное 0.

Далее решаем задачу коммивояжера для полученной матрицы методом ветвей и границ. Приведем алгоритм, состоящий из 9 шагов (в пунктирной рамке), поясняя выполнение каждого шага на примере.

1. По матрице W_0 получим приведенную матрицу W_0^*. Для этого из каждой строки матрицы W_0 вычтем минимальный элемент этой строки, а затем из каждого столбца полученной матрицы вычтем минимальный элемент этого столбца. В **приведенной матрице** в каждой ее строке и каждом ее столбце должен быть хотя бы один элемент, равный нулю. Полагаем $t=0$ (переменная t отвечает за нумерацию матриц).

Шаг 1 в нашем примере:

$$
\begin{array}{c} \\ \\ \\ \\ W_0= \\ \\ \\ \\ \end{array}
\begin{array}{c|ccccccc|c}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & \\
\hline
1 & \infty & 1 & 1 & 1 & 3 & 1 & 9 & 1 \\
2 & 11 & \infty & 5 & 5 & 11 & 6 & 17 & 5 \\
3 & 13 & 7 & \infty & 7 & 10 & 7 & 16 & 7 \\
4 & 12 & 5 & 5 & \infty & 10 & 5 & 16 & 5 \\
5 & 4 & 3 & 3 & 3 & \infty & 3 & 9 & 3 \\
6 & 13 & 7 & 5 & 6 & 14 & \infty & 20 & 5 \\
7 & 0 & 0 & 0 & 0 & 0 & 0 & \infty &
\end{array}
$$

Выпишем для каждой строки ее минимальный элемент:
1 - минимальный элемент 1-й строки;
5 - минимальный элемент 2-й строки;
7 - минимальный элемент 3-й строки;
5 - минимальный элемент 4-й строки;
3 - минимальный элемент 5-й строки;
5 - минимальный элемент 6-й строки.

Если минимальным элементом строки (столбца) является ноль, то его вычитание не изменит строку (столбец). Поэтому будем выписывать только ненулевые минимальные элементы. Вычитая из каждой строки ее минимальный элемент, из W_0 получим преобразованную матрицу:

$$
\begin{array}{c|ccccccc|}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & \infty & 0 & 0 & 0 & 2 & 0 & 8 \\
2 & 6 & \infty & 0 & 0 & 6 & 1 & 12 \\
3 & 6 & 0 & \infty & 0 & 3 & 0 & 9 \\
4 & 7 & 0 & 0 & \infty & 5 & 0 & 11 \\
5 & 1 & 0 & 0 & 0 & \infty & 0 & 6 \\
6 & 8 & 2 & 0 & 1 & 9 & \infty & 15 \\
7 & 0 & 0 & 0 & 0 & 0 & 0 & \infty \\
\hline
 & & & & & & & 6
\end{array}
$$

Далее выпишем для каждого столбца его минимальный элемент. При этом имеет смысл рассматривать только те столбцы, которые не содержат нулевых элементов. У нас таким является последний столбец. Вычитая из последнего столбца его минимальный элемент (6), получим приведенную матрицу W_0^* (записана справа).

$$
W_0^* =
\begin{array}{c|ccccccc|}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & \infty & 0 & 0 & 0 & 2 & 0 & 2 \\
2 & 6 & \infty & 0 & 0 & 6 & 1 & 6 \\
3 & 6 & 0 & \infty & 0 & 3 & 0 & 3 \\
4 & 7 & 0 & 0 & \infty & 5 & 0 & 5 \\
5 & 1 & 0 & 0 & 0 & \infty & 0 & 0 \\
6 & 8 & 2 & 0 & 1 & 9 & \infty & 9 \\
7 & 0 & 0 & 0 & 0 & 0 & 0 & \infty
\end{array}
$$

2. Сумма всех вычтенных чисел даст нижнюю оценку C_0 общего времени обработки деталей на линии. В процессе решения задачи эта величина может только увеличиться или остаться прежней.

Шаг 2 в нашем примере: $C_0 = \underbrace{1+5+7+5+3+5}_{\text{то, что вычли по строкам}} + \underbrace{6}_{\text{то, что вычли по столбцам}} = 32.$

3. Если размерность приведенной матрицы W_t^* равна 2-м, то работа алгоритма заканчивается. Записываем найденный цикл: в последовательность обработки деталей, полученную на предыдущих шагах, добавляем еще 2 недостающих перехода, соответствующих нулевым элементам матрицы W_t^*. Затем исключаем из цикла фиктивную деталь. Цикл при этом окажется разорванным, и вместо него мы получим искомую последовательность горячей обработки деталей за минимальное время, равное величине C_t. Последовательность начнется с той детали, которая следует непосредственно за фиктивной, и закончится той деталью, которая предшествовала фиктив-

ной детали в цикле. Порядок обработки деталей внутри последовательности будет таким же, как и в разорванном цикле. Далее, с учетом найденного порядка обработки, строим компактное расписание обработки деталей. При этом по матрице W_0 можно определять время w_{ij}, которое проходит от начала обработки детали i до начала обработки следующей за ней детали j на первом станке. Длина такого компактного расписания должна равняться C_t - минимальному времени, за которое можно обработать всю партию деталей на линии горячей обработки.

Если размерность приведенной матрицы W_t^* больше 2-х, то в W_t^* для каждого нуля подсчитаем его степень. Степень нуля равна сумме минимального элемента строки, в которой находится ноль, и минимального элемента столбца, в котором находится ноль. При выборе минимального элемента сам нулевой элемент, для которого вычисляется степень, не учитывается. ВЫЧИСЛЯТЬ СТЕПЕНИ НУЛЕЙ МОЖНО ТОЛЬКО В ПРИВЕДЕННЫХ МАТРИЦАХ!

Шаг 3 в нашем примере. Размерность W_0^* больше двух, поэтому вычисляем степени нулей.

$$W_0^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \left[\begin{array}{ccccccc} \infty & 0^0 & 0^0 & 0^0 & 2 & 0^0 & 2 \\ 6 & \infty & 0^0 & 0^0 & 6 & 1 & 6 \\ 6 & 0^0 & \infty & 0^0 & 3 & 0^0 & 3 \\ 7 & 0^0 & 0^0 & \infty & 5 & 0^0 & 5 \\ 1 & 0^0 & 0^0 & 0^0 & \infty & 0^0 & 0^2 \\ 8 & 2 & 0^1 & 1 & 9 & \infty & 9 \\ 0^1 & 0^0 & 0^0 & 0^0 & 0^2 & 0^0 & \infty \end{array}\right] \end{array}$$

Поясним на примерах, каким образом вычисляются степени нулей. Степень нуля, стоящего в 6-й строке и 3-м столбце, равна 1+0=1,т.к. минимальным элементом 6й строки (кроме рассматриваемого нуля) является 1, а *минимальным элементом 3го столбца* (кроме рассматриваемого нуля) является 0 (любой из оставшихся в этом столбце). Степень нуля, находящегося в 1й строке и 2м столбце равна 0+0=0, т.к., кроме рассматриваемого нуля, и в 1-й строке, и во 2-м столбце есть еще нулевые элементы, минимальные в этих строке и столбце.

4. Выбираем ноль с наибольшей степенью. Если таких нулей несколько, выбираем любой из них. Пусть выбранный ноль находится в строке i и столбце j. Тогда множество всех решений разбивается на два непересекающихся подмножества: в одном сразу после обработки детали i следует обработка детали j ($i \to j$), а в другом такая последовательность обработки запрещена ($i \not\to j$).

Шаг 4 в нашем примере: из нескольких нулей с одинаковой наибольшей степенью, равной двум, выберем ноль на пересечении 7-й строки и 5-го столбца. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход 7→5, а второе запрещает этот переход: $7 \not\to 5$.

5. Для случая $i \to j$ строим из приведенной матрицы W_t^* новую матрицу W_H следующим образом: в приведенной матрице W_t^* удаляем строку i и столбец j. При этом в новой матрице W_H нужно запретить переход, образующий цикл, в котором не все детали участвуют (для этого соответствующий элемент матрицы следует заменить знаком ∞). Возможные ситуации:

- Вновь добавляемый переход $i \to j$ не имеет общих точек с переходами, добавленными ранее (на предыдущих шагах). Тогда (для исключения цикла) элемент, стоящий в строке j и столбце i, заменяем знаком ∞.
- Вновь добавляемый переход $i \to j$ имеет общие вершины с переходами, добавленными ранее. Тогда соединяем переход с уже существующими переходами по общим вершинам - либо так: из $\boxed{j' \ldots \to i}$ и $\boxed{i \to j}$ получаем $\boxed{j' \ldots \to i \to j}$ (соединяем до i);

либо так: из $\boxed{i \to \dots i'}$ и $\boxed{i \to j}$ получаем $\boxed{i \to j \to \dots i'}$ (соединяем по j);

либо так: из $\boxed{i \to \dots i'}$, $\boxed{j' \dots \to i}$ и $\boxed{i \to j}$ получим $\boxed{j' \dots \to i \to j \to \dots i'}$ (соединив по i и j).

Для исключения цикла делаем запрет по концам цепочки, в которую добавили переход $i \to j$, в направлении, противоположном стрелкам. Укажем для трех случаев, рассмотренных выше, какой элемент матрицы заменяется знаком ∞:

для цепочки вида $\boxed{j' \dots \to i \to j}$ заменяем знаком ∞ элемент, стоящий в строке j и столбце j';

для цепочки вида $\boxed{i \to j \to \dots i'}$ заменяем знаком ∞ элемент, стоящий в строке i' и столбце i;

для цепочки вида $\boxed{j' \dots \to i \to j \to \dots i'}$ заменяем знаком ∞ элемент, стоящий в строке i' и столбце j'.

6. Для случая $i \not\to j$ строим из приведенной матрицы W_t^* новую матрицу W_{t2} следующим образом: в приведенной матрице W_t^* запрещаем переход из i в j, для этого элемент, стоящий в строке i и столбце j, заменяем знаком ∞.

7→5

$$W_{01} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ \\ \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 6 & 7 \end{array} \\ \left[\begin{array}{cccccc} \infty & 0 & 0 & 0 & 0 & 2 \\ 6 & \infty & 0 & 0 & 1 & 6 \\ 6 & 0 & \infty & 0 & 0 & 3 \\ 7 & 0 & 0 & \infty & 0 & 5 \\ 1 & 0 & 0 & 0 & 0 & \infty \\ 8 & 2 & 0 & 1 & \infty & 9 \end{array}\right] \\ \begin{array}{cccccc} 1 & & & & & 2 \end{array} \end{array}$$

В нашем примере выполняем шаги 5 и 6 алгоритма:

Шаг 5. В матрице W_{01}, полученной из W_0^* вычеркиванием 7-й строки и 5-го столбца, запрещаем знаком ∞ переход из 5 в 7 (по концам дуги 7→5 в противоположном направлении, чтобы не дать возможности в дальнейшем образоваться циклу, который не содержит обозначения всех 7 деталей).

7↛5

$$W_{02} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \left[\begin{array}{ccccccc} \infty & 0 & 0 & 0 & 2 & 0 & 2 \\ 6 & \infty & 0 & 0 & 6 & 1 & 6 \\ 6 & 0 & \infty & 0 & 3 & 0 & 3 \\ 7 & 0 & 0 & \infty & 5 & 0 & 5 \\ 1 & 0 & 0 & 0 & \infty & 0 & 0 \\ 8 & 2 & 0 & 1 & 9 & \infty & 9 \\ 0 & 0 & 0 & 0 & \infty & 0 & \infty \end{array}\right] \\ \begin{array}{ccccccc} & & & & 2 & & \end{array} \end{array}$$

Шаг 6. В матрице W_{02}, полученной переписыванием W_0^*, запрещаем знаком ∞ переход из 7 в 5 (т.к. рассматривается ситуация, когда переход из 7 в 5 запрещен).

7. По матрице W_{t1} строим соответствующую ей приведенную матрицу W_{t1}^* (см. шаг 1 алгоритма). Сумму всех вычтенных чисел добавляем к предыдущей оценке C_t. Получим оценку C_{t1}, которая показывает нижнюю оценку общего времени обработки деталей в случае, если после детали i обрабатывается деталь j.
8. По матрице W_{t2} строим соответствующую ей приведенную матрицу W_{t2}^* (см. шаг 1 алгоритма). Сумму всех вычтенных чисел добавляем к предыдущей оценке C_t. Получим оценку C_{t2}, которая показывает нижнюю оценку общего времени обработки деталей в случае, если после детали i не обрабатывается деталь j.
9. Рассматриваем оценки C_{t1}, C_{t2}, а при $t>1$ – также и те оценки, которые были получены на предыдущих шагах, но к данному моменту еще не выбирались. Из всех этих оценок выбираем наименьшую. Далее будем работать с матрицей, по которой получена эта наименьшая оценка (т.к. как нас интересует минимальное время обработки). Если имеется несколько <u>одинаковых</u> наименьших оценок, то из них выбираем ту, которая соответствует матрице, имеющей меньшую размерность. Выбранную матрицу обозначаем W_{t+1}^*. Соответствующую ей оценку обозначаем C_{t+1}. Полагаем $t := t+1$ (увеличиваем t на единицу) и переходим к шагу 3 алгоритма.

Выполним шаги 7, 8 и 9 алгоритма в нашем примере.

Из матриц W_{01} и W_{02} требуется получить соответствующие им приведенные.

Шаг 7. Матрица W_{01} содержит нули в каждой из строк, но в столбцах с номерами 1 и 7 нет нулевых элементов, поэтому, чтобы получить приведенную матрицу W_{01}^*, необходимо вычесть из столбца с номером 1 его минимальный элемент, равный 1, а из столбца с номером 7 – его минимальный элемент, равный 2. Сумму всех вычтенных чисел (1+2) добавим к предыдущей оценке C_0, равной 32. Получим оценку $C_{01}=32+1+2=35$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если сразу же после детали с номером 7 будет обрабатываться деталь с номером 5 ($7 \to 5$).

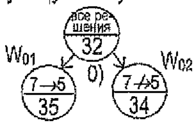
Шаг 8. Матрица W_{02} содержит нули в каждой из строк, но в столбце с номером 5 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{02}^*, необходимо вычесть из столбца с номером 5 его минимальный элемент, равный 2. Вычтенное число (2) добавим к предыдущей оценке C_0, равной 32. Получим оценку $C_{02}=32+2=34$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 5 непосредственно после детали с номером 7 ($7 \not\to 5$).

Приведенные матрицы W_{01}^* и W_{02}^* будут иметь вид, показанный ниже.

$7 \to 5$

$$W_{01}^* = \begin{array}{c|cccccc} & 1 & 2 & 3 & 4 & 6 & 7 \\ \hline 1 & \infty & 0 & 0 & 0 & 0 & 0 \\ 2 & 5 & \infty & 0 & 0 & 1 & 4 \\ 3 & 5 & 0 & \infty & 0 & 0 & 1 \\ 4 & 6 & 0 & 0 & \infty & 0 & 3 \\ 5 & 0 & 0 & 0 & 0 & 0 & \infty \\ 6 & 7 & 2 & 0 & 1 & \infty & 7 \end{array}$$

Чтобы отобразить процесс разбиения множества всех решений на подмножества, будем строить дерево решений, указывая в каждой вершине разрешенный или запрещенный переход и соответствующую ему нижнюю оценку времени.



$7 \not\to 5$

$$W_{02}^* = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & \infty & 0 & 0 & 0 & 0 & 0 & 2 \\ 2 & 6 & \infty & 0 & 0 & 4 & 1 & 6 \\ 3 & 6 & 0 & \infty & 0 & 1 & 0 & 3 \\ 4 & 7 & 0 & 0 & \infty & 3 & 0 & 5 \\ 5 & 1 & 0 & 0 & 0 & \infty & 0 & 0 \\ 6 & 8 & 2 & 0 & 1 & 7 & \infty & 9 \\ 7 & 0 & 0 & 0 & 0 & \infty & 0 & \infty \end{array}$$

Шаг 9. Из двух оценок $C_{01}=35$ и $C_{02}=34$ выбираем меньшую (т.к. нас интересует минимальное время обработки всех деталей). Это оценка $C_{02}=34$. Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{02}^*. Эту матрицу обозначаем W_1^*. Оценку C_{02} обозначим как C_1, т.е. $C_1=34$. $t := 0+1=1$. Возаращаемся к шагу 3 алгоритма.

$$W_1^* = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & \infty & 0^0 & 0^0 & 0^0 & 0^1 & 0^0 & 2 \\ 2 & 6 & \infty & 0^0 & 0^0 & 4 & 1 & 6 \\ 3 & 6 & 0^0 & \infty & 0^0 & 1 & 0^0 & 3 \\ 4 & 7 & 0^0 & 0^0 & \infty & 3 & 0^0 & 5 \\ 5 & 1 & 0^0 & 0^0 & 0^0 & \infty & 0^0 & 0^2 \\ 6 & 8 & 2 & 0^1 & 1 & 7 & \infty & 9 \\ 7 & 0^1 & 0^0 & 0^0 & 0^0 & \infty & 0^0 & \infty \end{array}$$

Шаг 3. Размерность матрицы W_1^* больше двух, поэтому вычисляем степени нулей этой матрицы.

Шаг 4. Выбираем ноль с наибольшей степенью.

Единственный нулевой элемент с максимальной степенью, равной 2, находится в 5-й строке и 7-м столбце. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход $5 \to 7$ (при условии, что переход из 7 в 5 запрещен), а второе запрещает переход: $5 \not\to 7$ (при таком же условии: $7 \not\to 5$).

$7 \not\to 5,\ 5 \to 7$

$$W_{11}=\begin{array}{c|cccccc} & 1&2&3&4&5&6\\ \hline 1&\infty&0&0&0&0&0\\ 2&6&\infty&0&0&4&1\\ 3&6&0&\infty&0&1&0\\ 4&7&0&0&\infty&3&0\\ 6&8&2&0&1&7&\infty\\ 7&0&0&0&0&\infty&0 \end{array}$$

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_1^* найдем матрицы W_{11} (для $5 \to 7$) и W_{12} (для $5 \not\to 7$).

Шаг 5. В матрице W_{11}, которая получена из W_1^* вычеркиванием 5-й строки и 7-го столбца, знаком ∞ переход из 7 в 5 (по концам дуги $5 \to 7$ в противоположном направлении) должен быть запрещен. В 7-й строке и 5-м столбце W_{11} уже стоит ∞, так что это требование выполнено.

$7 \not\to 5,\ 5 \not\to 7$

$$W_{12}=\begin{array}{c|ccccccc} & 1&2&3&4&5&6&7\\ \hline 1&\infty&0&0&0&0&0&2\\ 2&6&\infty&0&0&4&1&6\\ 3&6&0&\infty&0&1&0&3\\ 4&7&0&0&\infty&3&0&5\\ 5&1&0&0&0&\infty&0&\infty\\ 6&8&2&0&1&7&\infty&9\\ 7&0&0&0&0&\infty&0&\infty \end{array}$$

2

Шаг 6. В матрице W_{12}, полученной переписыванием W_1^*, запрещаем знаком ∞ переход из 5 в 7 (т.к. рассматривается ситуация, когда переход из 5 в 7 запрещен).

Выполним далее для нашего примера шаги 7, 8 и 9 алгоритма, т.е. найдем приведенные матрицы для W_{11} и W_{12}, а также уточним оценку C_1 для каждой из них.

Шаг 7. Матрица W_{11} содержит нули в каждой из строк и в каждом из столбцов, поэтому она совпадает с соответствующей ей приведенной матрицей W_{11}^* (ничего вычитать не нужно). Поэтому предыдущую оценку C_1, равную 34, «увеличиваем» на 0 единиц. Получим оценку $C_{11}=34+0=34$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если сразу же после детали с номером 5 будет обрабатываться деталь с номером 7 ($5 \to 7$), при условии, что $7 \not\to 5$.

Шаг 8. Матрица W_{12} содержит нули в каждой из строк, но в столбце с номером 7 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{12}^*, необходимо вычесть из столбца с номером 7 его минимальный элемент, равный 2. Вычтенное число (2) добавим к предыдущей оценке C_1, равной 34. Получим оценку $C_{12}=34+2=36$. Она дает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали D_7 сразу после детали D_5 ($5 \not\to 7$), при условии, что $7 \not\to 5$.

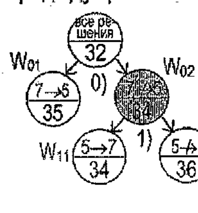
Приведенные матрицы W_{11}^* и W_{12}^* будут иметь вид, показанный ниже.

$7 \not\to 5,\ 5 \to 7$

$$W_{11}^*=\begin{array}{c|cccccc} & 1&2&3&4&5&6\\ \hline 1&\infty&0&0&0&0&0\\ 2&6&\infty&0&0&4&1\\ 3&6&0&\infty&0&1&0\\ 4&7&0&0&\infty&3&0\\ 6&8&2&0&1&7&\infty\\ 7&0&0&0&0&\infty&0 \end{array}$$

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{02}), выбранной на предыдущем этапе.

На рисунке эта вершина закрашена.



$7 \not\to 5,\ 5 \not\to 7$

$$W_{12}^*=\begin{array}{c|ccccccc} & 1&2&3&4&5&6&7\\ \hline 1&\infty&0&0&0&0&0&0\\ 2&6&\infty&0&0&4&1&4\\ 3&6&0&\infty&0&1&0&1\\ 4&7&0&0&\infty&3&0&3\\ 5&1&0&0&0&\infty&0&\infty\\ 6&8&2&0&1&7&\infty&7\\ 7&0&0&0&0&\infty&0&\infty \end{array}$$

Шаг 9. Из оценок $C_{11}=34$ и $C_{12}=36$, а также $C_{01}=35$ (заметим, что рассматриваемые оценки соответствуют висячим вершинам дерева, из которых не построены ветви) выбираем наименьшую. Это оценка $C_{11}=34$. Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{11}^*. Эту матрицу обозначаем W_2^*. Оценку C_{11} обозначим как C_2, т.е. $C_2=34$. $t := 1+1=2$. Переходим к шагу 3 алгоритма.

$$W_2^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} \infty & 0^0 & 0^0 & 0^0 & 0^1 & 0^0 \\ 6 & \infty & 0^0 & 0^0 & 4 & 1 \\ 6 & 0^0 & \infty & 0^0 & 1 & 0^0 \\ 7 & 0^0 & 0^0 & \infty & 3 & 0^0 \\ 8 & 2 & 0^1 & 1 & 7 & \infty \\ 0^6 & 0^0 & 0^0 & 0^0 & \infty & 0^0 \end{bmatrix} \end{array}$$

Шаг 3. Размерность матрицы W_2^* больше двух, поэтому вычисляем степени нулей этой матрицы.

Шаг 4. Выбираем ноль с наибольшей степенью. Единственный ноль с максимальной степенью, равной 6, находится в 7-й строке и 1-м столбце. Тогда множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход 7→1 (при этом 7↛5 и 5→7), а второе запрещает этот переход: 7↛1 (при тех же условиях: 7↛5 и 5→7).

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_2^* найдем матрицы W_{21} (для 7→1) и W_{22} (для 7↛1).

7↛5,5→7→1

$$W_{21} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & 0 & 4 & 1 \\ 0 & \infty & 0 & 1 & 0 \\ 0 & 0 & \infty & 3 & 0 \\ 2 & 0 & 1 & 7 & \infty \end{bmatrix} \\ \begin{array}{ccccc} & & & 1 & \end{array} \end{array}$$

Шаг 5. В матрице W_{21}, которая получена из W_2^* вычеркиванием 7-й строки и 1-го столбца, запрещаем переход из 1 в 5: т.к. добавляемый переход 7→1 имеет общую вершину **7** с переходом 5→7, добавленным ранее, то, *соединив их по этой общей вершине*, получим цепочку 5→7→1, по концам которой из 1 в 5 запрещаем переход, чтобы не образовался цикл, содержащий не все детали.

7↛5, 5→7, 7↛1

$$W_{22} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ 7 \\ \\ \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} \infty & 0 & 0 & 0 & 0 & 0 \\ 6 & \infty & 0 & 0 & 4 & 1 \\ 6 & 0 & \infty & 0 & 1 & 0 \\ 7 & 0 & 0 & \infty & 3 & 0 \\ 8 & 2 & 0 & 1 & 7 & \infty \\ \infty & 0 & 0 & 0 & \infty & 0 \end{bmatrix} \\ \begin{array}{cccccc} 6 & & & & & \end{array} \end{array}$$

Шаг 6. В матрице W_{22}, полученной переписыванием W_2^*, запрещаем переход из 7 в 1. Выполним далее для нашего примера шаги 7, 8 и 9 алгоритма, т.е. найдем приведенные матрицы для W_{21} и W_{22}, а также уточним оценку C_2 для каждой из них.

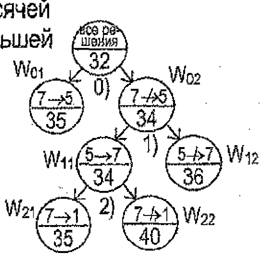
Шаг 7. Матрица W_{21} содержит нули в каждой строке, но в столбце номер 5 нет нулей, поэтому, чтобы получить приведенную матрицу W_{21}^*, вычитаем из столбца с номером 5 его минимальный элемент. Вычтенное число (1) добавим к предыдущей оценке C_2, равной 34. Получим оценку C_{21}=34+1=35 – нижнюю оценку общего времени обработки всех деталей, если сразу после D_7 обрабатывать D_1 (7→1), при этом 7↛5 и 5→7.

Шаг 8. Матрица W_{22} содержит нули в каждой из строк, но в столбце с номером 1 нет нулей, поэтому, чтобы получить приведенную матрицу W_{22}^*, надо вычесть из столбца номер 1 его минимальный элемент, равный 6. Вычтенное число (6) добавим к предыдущей оценке C_2, равной 34. Получим оценку C_{22}=34+6=40. Она дает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали D_1 непосредственно после детали D_7 (7↛1), при условии, что 7↛5 и 5→7.

7↛5,5→7→1

$$W_{21}^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & 0 & 3 & 1 \\ 0 & \infty & 0 & 0 & 0 \\ 0 & 0 & \infty & 2 & 0 \\ 2 & 0 & 1 & 6 & \infty \end{bmatrix} \end{array}$$

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{11}), выбранной на предыдущем этапе.



7↛5, 5→7, 7↛1

$$W_{22}^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{cccccc} 1 & 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} \infty & 0 & 0 & 0 & 0 & 0 \\ 0 & \infty & 0 & 0 & 4 & 1 \\ 0 & 0 & \infty & 0 & 1 & 0 \\ 1 & 0 & 0 & \infty & 3 & 0 \\ 2 & 2 & 0 & 1 & 7 & \infty \\ \infty & 0 & 0 & 0 & \infty & 0 \end{bmatrix} \end{array}$$

Шаг 9. Из оценок $C_{01}=35$, $C_{21}=35$, $C_{22}=40$, $C_{12}=36$, соответствующих висячим вершинам дерева, выбираем наименьшую. Наименьшей является любая из оценок $C_{21}=35$ или $C_{01}=35$. Из них выбираем ту, которая соответствует матрице меньшей размерности. Оценке $C_{01}=35$ соответствует матрица W_{01}^* размерности 7х7. Оценке $C_{21}=35$ соответствует матрица W_{21}^* размерности 5х5. (Заметим, что чем ниже ярус дерева, на котором находится оценка, тем меньше размерность матрицы, соответствующей этой оценке.) Выбираем оценку $C_{21}=35$. Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{21}^*. Эту матрицу обозначаем W_3^*. Оценку C_{21} обозначим как C_3, т.е. $C_3=35$. t := 2+1=3. Переходим к шагу 3 алгоритма.

$$W_3^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \left[\begin{array}{ccccc} 0^0 & 0^0 & 0^0 & \infty & 0^0 \\ \infty & 0^0 & 0^0 & 3 & 1 \\ 0^0 & \infty & 0^0 & 0^2 & 0^0 \\ 0^0 & 0^0 & \infty & 2 & 0^0 \\ 2 & 0^1 & 1 & 6 & \infty \end{array}\right] \end{array}$$

Шаг 3. Размерность W_3^* больше двух, поэтому ищем степени нулей.

Шаг 4. Единственный нулевой элемент с максимальной степенью, равной 2, находится в 3-й строке и 5-м столбце. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход $3\to5$ (при условии, что $7\not\to5$ и $5\to7\to1$), а второе запрещает этот переход: $3\not\to5$ (при тех же условиях).

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_3^* найдем матрицы W_{31} (для $3\to5$) и W_{32} (для $3\not\to5$).

$7\not\to5, 3\to5\to7\to1$

$$W_{31} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \left[\begin{array}{cccc} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 2 & 0 & 1 & \infty \end{array}\right] \end{array}$$

Шаг 5. В матрице W_{31}, которая получена из W_3^* вычеркиванием 3-й строки и 5-го столбца, знаком ∞ запрещаем переход из 1 в 3. Т.к. добавляемый переход $3\to5$ имеет общую вершину 5 с цепочкой $5\to7\to1$, полученной ранее, то, соединив их по этой общей вершине, получим цепочку $3\to5\to7\to1$, по концам которой из 1 в 3 запрещаем переход, чтобы не образовался цикл, содержащий не все детали.

$7\not\to5$, $5\to7\to1$, $3\not\to5$

$$W_{32} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \left[\begin{array}{ccccc} 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & 0 & 3 & 1 \\ 0 & \infty & 0 & \infty & 0 \\ 0 & 0 & \infty & 2 & 0 \\ 2 & 0 & 1 & 6 & \infty \end{array}\right] \\ \begin{array}{ccccc} & & & 2 & \end{array} \end{array}$$

Шаг 6. В матрице W_{32}, полученной переписыванием W_3^*, запрещаем переход из 3 в 5. Выполним далее для нашего примера шаги 7, 8 и 9 алгоритма, т.е. найдем приведенные матрицы для W_{31} и W_{32}, а также уточним оценку C_3 для каждой из них.

Шаг 7. Матрица W_{31} содержит нули в каждой из строк и в каждом из столбцов, поэтому совпадает со своей приведенной матрицей W_{31}^*. Соответствующая этой матрице оценка C_{31} совпадает с предыдущей оценкой C_3, равной 35, и дает нижнюю оценку общего времени обработки всех деталей в случае, если сразу же после детали с номером 3 будет обрабатываться деталь с номером 5 ($3\to5$), при условии, что $7\not\to5$ и $5\to7\to1$.

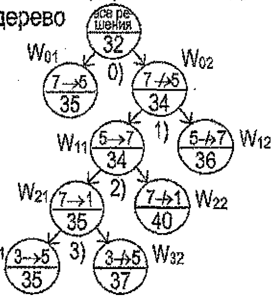
Шаг 8. Матрица W_{32} содержит нули в каждой из строк, но в столбце с номером 5 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{32}^*, необходимо вычесть из столбца с номером 5 его минимальный элемент, равный 2. Вычтенное число (2) добавим к предыдущей оценке C_3, равной 35. Получим оценку $C_{32}=35+2=37$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 5 непосредственно после детали с номером 3 ($3\not\to5$), при условии, что $7\not\to5$ и $5\to7\to1$.

Приведенные матрицы W_{31}^* и W_{32}^* будут иметь вид, показанный ниже.

$7 \not\to 5$, $\mathbf{3 \to 5 \to 7 \to 1}$

$$W_{31}^* = \begin{array}{c|c} & \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 4 \\ 6 \end{array} & \begin{bmatrix} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 2 & 0 & 1 & \infty \end{bmatrix} \end{array}$$

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{21}), выбранной на предыдущем этапе.



$7 \not\to 5$, $5 \to 7 \to 1$, $\mathbf{3 \not\to 5}$

$$W_{32}^* = \begin{array}{c|c} & \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 6 \end{array} & \begin{bmatrix} 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & 0 & 1 & 1 \\ 0 & \infty & 0 & \infty & 0 \\ 0 & 0 & \infty & 0 & 0 \\ 2 & 0 & 1 & 4 & \infty \end{bmatrix} \end{array}$$

Шаг 9. Из оценок C_{01}=35, C_{31}=35, C_{32}=37, C_{22}=40, C_{12}=36, соответствующих висячим вершинам дерева, выбираем наименьшую. Это оценка C_{31}=35 – она наименьшая, как и C_{01}, но ей соответствует матрица меньшей размерности, чем у C_{01}. Далее будем работать с матрицей W_{31}^*, для которой получена эта оценка. Эту матрицу обозначаем W_4^*. Оценку C_{31} обозначим как C_4, т.е. C_4=35. t :=3+1=4. Переходим к шагу 3 алгоритма.

$$W_4^* = \begin{array}{c|c} & \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 4 \\ 6 \end{array} & \begin{bmatrix} 0^0 & \infty & 0^0 & 0^0 \\ \infty & 0^0 & 0^0 & 1 \\ 0^0 & 0^0 & \infty & 0^0 \\ 2 & 0 & 1 & \infty \end{bmatrix} \end{array}$$

Шаг 3. Размерность W_4^* больше двух, поэтому ищем степени нулей.

Шаг 4. Ноль с максимальной степенью, равной 1, есть в 6-й строке и 3-м столбце. Тогда мн-во всех решений разобьется на два непересекающихся подмножества: 1-е содержит переход $6 \to 3$ (при этом $7 \not\to 5$ и $3 \to 5 \to 7 \to 1$), 2-е при тех же условиях запрещает этот переход: $6 \not\to 3$.

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_4^* найдем матрицы W_{41} (для $6 \to 3$) и W_{42} (для $6 \not\to 3$).

$7 \not\to 5$,
$\mathbf{6 \to 3} \to 5 \to 7 \to 1$

$$W_{41} = \begin{array}{c|c} & \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & \infty \\ \infty & 0 & 1 \\ 0 & \infty & 0 \end{bmatrix} \end{array}$$

Шаг 5. В матрице W_{41}, которая получена из W_4^* вычеркиванием 6-й строки и 3-го столбца, запрещаем переход из 1 в 6. Т.к. добавляемый переход $6 \to 3$ имеет общую вершину 3 с цепочкой $3 \to 5 \to 7 \to 1$, полученной ранее, то, соединив их по вершине 3, получим цепочку $6 \to 3 \to 5 \to 7 \to 1$; по концам ее из 1 в 6 запрещаем переход, чтобы не получить цикл, не содержащий все детали.

$7 \not\to 5$, $3 \to 5 \to 7 \to 1$,
$\mathbf{6 \not\to 3}$

$$W_{42} = \begin{array}{c|c|c} & \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} & \\ \begin{array}{c} 1 \\ 2 \\ 4 \\ 6 \end{array} & \begin{bmatrix} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 2 & \infty & 1 & \infty \end{bmatrix} & \begin{array}{c} \\ \\ \\ 1 \end{array} \end{array}$$

Шаг 6. В матрице W_{42}, полученной переписыванием W_4^*, запрещаем переход из 6 в 3. Выполним далее для нашего примера шаги 7, 8 и 9 алгоритма, т.е. найдем приведенные матрицы для W_{41} и W_{42}, а также уточним оценку C_4 для каждой из них.

Шаг 7. Матрица W_{41} содержит нули в каждой строке и в каждом столбце, поэтому совпадает со своей приведенной матрицей W_{41}^*. Соответствующая этой матрице оценка C_{41} совпадает с предыдущей оценкой C_4, равной 35, и дает нижнюю оценку общего времени обработки всех деталей, если сразу же после детали 6 будет обрабатываться деталь 3 ($6 \to 3$), учитывая, что $3 \to 5 \to 7 \to 1$, $7 \not\to 5$.

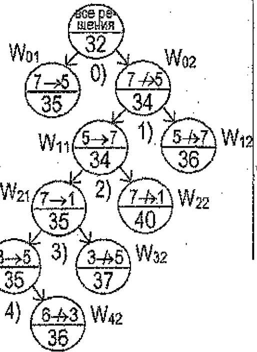
Шаг 8. Матрица W_{42} содержит нули в каждом из столбцов, но в строке с номером 6 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{42}^*, необходимо вычесть из строки с номером 6 ее минимальный элемент, равный 1. Вычтенное число (1) добавим к предыдущей оценке C_4, равной 35. Получим оценку C_{42}=35+1=36, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 3 непосредственно после детали с номером 6 (6↛3), при условии, что 7↛5 и 3→5→7→1.

7↛5,

6→3→5→7→1

$$W_{41}^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \left[\begin{array}{ccc} 0 & 0 & \infty \\ \infty & 0 & 1 \\ 0 & \infty & 0 \end{array}\right] \end{array}$$

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{31}), выбранной на предыдущем этапе.



7↛5, 3→5→7→1,

6↛3

$$W_{42}^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ 6 \end{array} \begin{array}{c} \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \left[\begin{array}{cccc} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 1 & \infty & 0 & \infty \end{array}\right] \end{array}$$

Шаг 9. Из оценок C_{01}=35, C_{41}=35, C_{42}=36, C_{32}=37, C_{22}=40, C_{12}=36 выбираем наименьшую. Это оценка C_{41}=35 (соответствующая матрице размерности 3х3, меньшей, чем размерность матрицы. соответствующей оценке C_{01}). Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{41}^*. Ее обозначаем W_5^*. Оценку C_{41} обозначим как C_5, т.е. C_5=35. t := 4+1=5. Переходим к шагу 3 алгоритма.

$$W_5^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \left[\begin{array}{ccc} 0^0 & 0^0 & \infty \\ \infty & 0^1 & 1 \\ 0^0 & \infty & 0^1 \end{array}\right] \end{array}$$

Шаг 3. Размерность W_5^* больше двух, поэтому ищем степени нулей.

Шаг 4. Из двух нулевых элементов с максимальной степенью, равной 1, выберем любой - например, тот, который находится во 2-й строке и 4-м столбце. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход 2→4 (при условии, что 7↛5 и 6→3→5→7→1), а второе запрещает этот переход: 2↛4 (при тех же условиях: 7↛5 и 6→3→5→7→1).

Выполним шаги 5 и 6, т.е. по W_5^* найдем матрицы W_{51} (для 2→4) и W_{52} (для 2↛4).

7↛5, **2→4**,

6→3→5→7→1

$$W_{51} = \begin{array}{c} \\ 1 \\ 4 \end{array} \begin{array}{c} \begin{array}{cc} 2 & 6 \end{array} \\ \left[\begin{array}{cc} 0 & \infty \\ \infty & 0 \end{array}\right] \end{array}$$

Шаг 5. В матрице W_{51}, полученной из W_5^* вычеркиванием 2-й строки и 4-го столбца, запрещаем переход из 4 в 2. Т.к. добавляемый переход 2→4 не имеет общих вершин с полученной ранее цепочкой 6→3→5→7→1, то запрещаем переход по концам вновь добавляемой дуги 2→4 в противоположном направлении, из 4 в 2.

7↛5, **2↛4**,

6→3→5→7→1

$$W_{52} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \left[\begin{array}{ccc} 0 & 0 & \infty \\ \infty & \infty & 1 \\ 0 & \infty & 0 \end{array}\right] \end{array} \begin{array}{c} \\ \\ 1 \\ \\ \end{array}$$

Шаг 6. В матрице W_{52}, полученной переписыванием W_5^*, запрещаем переход из 2 в 4. Выполним далее для нашего примера шаги 7, 8 и 9 алгоритма, т.е. найдем приведенные матрицы для W_{51} и W_{52}, и уточним оценку C_5 для каждой из этих матриц.

Шаг 7. Матрица W_{51} содержит нули в каждой строке и в каждом столбце, поэтому совпадает со своей приведенной матрицей W_{51}^*. Соответствующая этой матрице оценка C_{51} совпадает с предыдущей оценкой C_5, равной 35, и показывает общее время обработки всех деталей, если сразу же после детали номер 2 будет обрабатываться деталь номер 4 (2→4), при этом 7↛5 и 6→3→5→7→1.

Шаг 8. Матрица W_{52} содержит нули в каждом из столбцов, но в строке с номером 2 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{52}^*, необходимо вычесть из строки с номером 2 ее минимальный элемент, равный 1. Получим оценку C_{52}=35+1=36, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 4 непосредственно после детали с номером 2 (2↛4), при том, что 7↛5 и 6→3→5→7→1.

Приведенные матрицы W_{51}^* и W_{52}^* будут иметь вид:

2→4,
6→3→5→7→1

$$W_{51}^* = \begin{array}{c c} & \begin{array}{cc} 2 & 6 \end{array} \\ \begin{array}{c} 1 \\ 4 \end{array} & \begin{bmatrix} 0 & \infty \\ \infty & 0 \end{bmatrix} \end{array}$$

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{41}), выбранной на предыдущем этапе.

6→3→5→7→1,
2↛4

$$W_{52}^* = \begin{array}{c c} & \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 4 \end{array} & \begin{bmatrix} 0 & 0 & \infty \\ \infty & \infty & 0 \\ 0 & \infty & 0 \end{bmatrix} \end{array}$$

Шаг 9. Из оценок C_{01}=35, C_{51}=35, C_{52}=36, C_{42}=36, C_{32}=37, C_{22}=40, C_{12}=36 выбираем наименьшую. Это оценка C_{51}=35 (т.к. ей соответствует матрица размерности меньшей, чем у C_{01}). Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{51}^*. Ее обозначаем W_6^*. Оценку C_{51} обозначим как C_6, т.е. C_6=35. t := 5+1=6. Переходим к шагу 3 алгоритма.

Шаг 3. Размерность W_6^* равна двум. На этом работа алгоритма завершена.

2→4, 6→3→5→7→1

$$W_6^* = \begin{array}{c c} & \begin{array}{cc} 2 & 6 \end{array} \\ \begin{array}{c} 1 \\ 4 \end{array} & \begin{bmatrix} 0 & \infty \\ \infty & 0 \end{bmatrix} \end{array}$$

По матрице W_6^* видно, что в ней разрешены как раз те переходы, которых не хватает для получения из цепочек 2→4 и 6→3→5→7→1 полного цикла: 1→2 и 4→6. Т.к. в этой матрице нулевые элементы имеют одинаковую степень, равную 0, то выбираем любой из них, например, стоящий в 4-й строке и 6-м столбце. При добавлении перехода 4→6 к построенным ранее цепочкам 2→4 и 6→3→5→7→1 получим следующий результат: 2→4→6→3→5→7→1. После вычеркивания 4-й строки и 6-го столбца в матрице останется единственный элемент (в 1-й строке и 2-м столбце). Здесь не требуется делать запрет знаком ∞ по концам полученной цепочки, т.к. к этому моменту все 7 деталей участвуют в образовании цепочки ровно по одному разу. Таким образом, добавляем недостающий переход (1→2) и получаем цикл, в котором участвуют все детали:

2→4→6→3→5→7→1

При этом полное минимальное время обработки этих деталей в указанном порядке составляет C_6=35 единиц. Проверим это, используя исходную матрицу W_0. Каждому переходу i→j в найденном цикле поставлено в соответствие число – элемент w_{ij} матрицы W_0.

$$2 \xrightarrow{5} 4 \xrightarrow{5} 6 \xrightarrow{5} 3 \xrightarrow{10} 5 \xrightarrow{9} 7 \xrightarrow{0} 1$$
1

Найдем сумму этих чисел: 5 + 5 + 5 + 10 + 9 + 0 + 1 = 35 – проверка подтвердила, что общее время обработки получено правильно.

Но в полученном цикле присутствует фиктивная 7-я деталь. Удалив ее из цикла вместе с стоящими непосредственно слева и справа от нее стрелками (цикл при этом разорвется), получаем окончательно искомый порядок обработки заданных шести деталей за минимальное время, равное 35: $2\to4\to6\to3\to5\to\cancel{7}\to1$.

$D_1\to D_2\to D_4\to D_6\to D_3\to D_5$ – найденный порядок обработки.

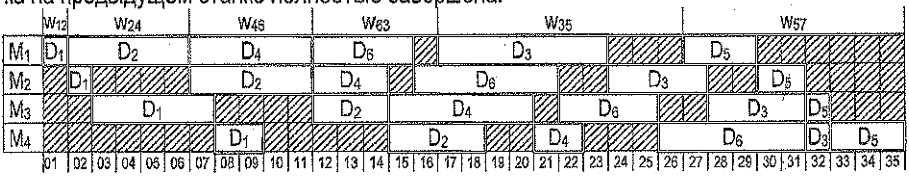
Построим компактное расписание, соответствующее найденному оптимальному порядку обработки. Для этого обратимся к матрицам P и W_0.

$$P = \begin{array}{c} \\ M_1 \\ M_2 \\ M_3 \\ M_4 \end{array} \begin{array}{c} \begin{array}{cccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 \end{array} \\ \begin{bmatrix} 1 & 5 & 7 & 5 & 3 & 4 \\ 1 & 5 & 4 & 3 & 2 & 6 \\ 5 & 3 & 4 & 6 & 1 & 4 \\ 2 & 4 & 1 & 2 & 3 & 6 \end{bmatrix} \end{array}$$

$$W_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \end{array} \begin{array}{c} \begin{array}{ccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 \end{array} \\ \begin{bmatrix} \infty & 1 & 1 & 1 & 3 & 1 & 9 \\ 11 & \infty & 5 & 5 & 11 & 6 & 17 \\ 13 & 7 & \infty & 7 & 10 & 7 & 16 \\ 12 & 5 & 5 & \infty & 10 & 5 & 16 \\ 4 & 3 & 3 & 3 & \infty & 3 & 9 \\ 13 & 7 & 5 & 6 & 14 & \infty & 20 \\ 0 & 0 & 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array}$$

По матрице P определяем время обработки каждой детали на каждом станке.

По матрице W_0 определяем время, которое проходит от начала обработки одной детали до начала обработки следующей за ней детали на первом станке. Имеем: $w_{12}=1$, $w_{24}=5$, $w_{46}=5$, $w_{63}=5$, $w_{35}=10$. При построении такой диаграммы учитываем, что обработка детали на очередном освободившемся станке начинается только после того, как ее обработка на предыдущем станке полностью завершена.



Время обработки по полученному расписанию действительно составляет 35 единиц.

<u>*Примечание.*</u> Приведем пример того, как величины w_{ij} рассчитываются по формуле:

$$w_{ij} = \max_{1\le h\le k}\left\{\sum_{s=1}^{h} p_{si} - \sum_{s=1}^{h-1} p_{sj}\right\},\ i,j = 1,2,\ldots,m.$$ Найдем, например, $$w_{63} = \max_{1\le h\le 4}\left\{\sum_{s=1}^{h} p_{s6} - \sum_{s=1}^{h-1} p_{s3}\right\}.$$

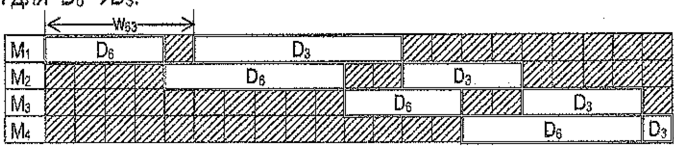
$$w_{63} = \max(\underbrace{p_{16} - 0}_{h=1},\ \underbrace{(p_{16}+p_{26}) - p_{13}}_{h=2},\ \underbrace{(p_{16}+p_{26}+p_{36}) - (p_{13}+p_{23})}_{h=3},\ \underbrace{(p_{16}+p_{26}+p_{36}+p_{46}) - (p_{13}+p_{23}+p_{33})}_{h=4}) =$$

$$= \max(4-0,\ (4+6)-7,\ (4+6+4)-(7+4),\ (4+6+4+6)-(7+4+4)) =$$

$$= \max(4,\ 3,\ 3,\ 5) =$$

= 5. Здесь h принимает значения от 1 до 4, где 4 – количество станков в нашей задаче.

Это же значение $w_{63} = 5$ можно получить, построив фрагмент диаграммы компактного расписания для $D_6\to D_3$:



2.2. Пример оформления решения задачи о составлении РГО

Примечание. *В данном Примере построение дерева решений показано поэтапно. В своей работе стройте* ***одно*** *дерево решений, постепенно наращивая его.*

Решение. Матрица P, задающая длительности обработки деталей каждым станком:

	D_1	D_2	D_3	D_4	D_5	D_6
M_1	1	5	7	5	3	4
M_2	1	5	4	3	2	6
M_3	5	3	4	6	1	4
M_4	2	4	1	2	3	6

Запишем матрицу величин w_{ij}, полученную по заданной матрице P в *Derive*, добавим в нее 7-ю строку и 7-й столбец, а элементы главной диагонали заменим знаком ∞:

$$W_0=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\7\\ \\ \end{array}\begin{array}{c}\begin{array}{ccccccc}1&2&3&4&5&6&7\end{array}\\ \left[\begin{array}{ccccccc}\infty&1&1&1&3&1&9\\11&\infty&5&5&11&6&17\\13&7&\infty&7&10&7&16\\12&5&5&\infty&10&5&16\\4&3&3&3&\infty&3&9\\13&7&5&6&14&\infty&20\\0&0&0&0&0&0&\infty\end{array}\right]\\ \begin{array}{ccccccc}&&&&&&6\end{array}\end{array}\begin{array}{c}\\1\\5\\7\\5\\3\\5\\ \\ \\ \end{array}$$

Приведенная матрица W_0^*:

$$W_0^*=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\7\end{array}\begin{array}{c}\begin{array}{ccccccc}1&2&3&4&5&6&7\end{array}\\ \left[\begin{array}{ccccccc}\infty&0&0&0&2&0&2\\6&\infty&0&0&6&1&6\\6&0&\infty&0&3&0&3\\7&0&0&\infty&5&0&8\\1&0&0&0&\infty&0&0\\8&2&0&1&9&\infty&9\\0&0&0&0&0&0&\infty\end{array}\right]\end{array}$$

$C_0=1+5+7+5+3+5+6=32$

Строим корневую вершину дерева:

Находим степени нулей в W_0^* и выбираем 0 с наибольшей степенью:

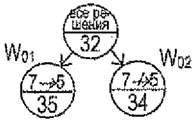
$$W_0^*=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\7\end{array}\begin{array}{c}\begin{array}{ccccccc}1&2&3&4&5&6&7\end{array}\\ \left[\begin{array}{ccccccc}\infty&0^0&0^0&0^0&2&0^0&2\\6&\infty&0^0&0^0&6&1&6\\6&0^0&\infty&0^0&3&0^0&3\\7&0^0&0^0&\infty&5&0^0&8\\1&0^0&0^0&0^0&\infty&0^0&0^2\\8&2&0^1&1&9&\infty&9\\0^1&0^0&0^0&0^0&0^2&0^0&\infty\end{array}\right]\end{array}$$

Рассматриваем 2 случая – **7→5** и **7-/->5**, и строим очередной ярус дерева решений:

7→5

$$W_{01}=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\ \\ \end{array}\begin{array}{c}\begin{array}{cccccc}1&2&3&4&6&7\end{array}\\ \left[\begin{array}{cccccc}\infty&0&0&0&0&2\\6&\infty&0&0&1&6\\6&0&\infty&0&0&3\\7&0&0&\infty&0&8\\1&0&0&0&0&\infty\\8&2&0&1&\infty&9\end{array}\right]\\ \begin{array}{cccccc}1&&&&&2\end{array}\end{array}$$

$C_{01}=32+1+2=35$



7-/->5

$$W_{02}=\begin{array}{c}\\1\\2\\3\\4\\5\\6\\7\\ \\ \end{array}\begin{array}{c}\begin{array}{ccccccc}1&2&3&4&5&6&7\end{array}\\ \left[\begin{array}{ccccccc}\infty&0&0&0&2&0&2\\6&\infty&0&0&6&1&6\\6&0&\infty&0&3&0&3\\7&0&0&\infty&5&0&8\\1&0&0&0&\infty&0&0\\8&2&0&1&9&\infty&9\\0&0&0&0&\infty&0&\infty\end{array}\right]\\ \begin{array}{ccccccc}&&&&2&&\end{array}\end{array}$$

$C_{02}=32+2=34$

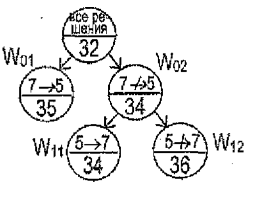
Т.к. оценка для матрицы W_{02} наименьшая, переходим к приведенной матрице W_{02}^*:

$$W_{02}^* = \begin{array}{c|ccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & \infty & 0^0 & 0^0 & 0^0 & 0^1 & 0^0 & 2 \\
2 & 6 & \infty & 0^0 & 0^0 & 4 & 1 & 6 \\
3 & 6 & 0^0 & \infty & 0^0 & 1 & 0^0 & 3 \\
4 & 7 & 0^0 & 0^0 & \infty & 3 & 0^0 & 8 \\
5 & \cancel{1} & \cancel{0^0} & \cancel{0^0} & \cancel{0^0} & \cancel{\infty} & \cancel{0^0} & 0^2 \\
6 & 8 & 2 & 0^1 & 1 & 7 & \infty & 9 \\
7 & 0^1 & 0^0 & 0^0 & 0^0 & \infty & 0^0 & \infty
\end{array}$$

Рассматриваем 2 случая – **5→7** и **5↛7**, и строим очередной ярус дерева решений:

$7 \not\to 5$, $\mathbf{5 \to 7}$

$$W_{11} = \begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & \infty & 0 & 0 & 0 & 0 & 0 \\
2 & 6 & \infty & 0 & 0 & 4 & 1 \\
3 & 6 & 0 & \infty & 0 & 1 & 0 \\
4 & 7 & 0 & 0 & \infty & 3 & 0 \\
6 & 8 & 2 & 0 & 1 & 7 & \infty \\
7 & 0 & 0 & 0 & 0 & \infty & 0
\end{array}$$



$7 \not\to 5$, $\mathbf{5 \not\to 7}$

$$W_{12} = \begin{array}{c|ccccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\
\hline
1 & \infty & 0 & 0 & 0 & 0 & 0 & 2 \\
2 & 6 & \infty & 0 & 0 & 4 & 1 & 6 \\
3 & 6 & 0 & \infty & 0 & 1 & 0 & 3 \\
4 & 7 & 0 & 0 & \infty & 3 & 0 & 8 \\
5 & 1 & 0 & 0 & 0 & \infty & 0 & \infty \\
6 & 8 & 2 & 0 & 1 & 7 & \infty & 9 \\
7 & 0 & 0 & 0 & 0 & \infty & 0 & \infty \\
\hline
 & & & & & & & 2
\end{array}$$

$C_{11}=34+0=34$

$C_{12}=34+2=36$

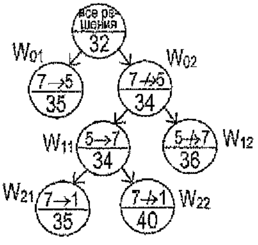
Т.к. оценка для матрицы W_{11} наименьшая, переходим к приведенной матрице W_{11}^*:

$$W_{11}^* = \begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & \infty & 0^0 & 0^0 & 0^0 & 0^1 & 0^0 \\
2 & 6 & \infty & 0^0 & 0^0 & 4 & 1 \\
3 & 6 & 0^0 & \infty & 0^0 & 1 & 0^0 \\
4 & 7 & 0^0 & 0^0 & \infty & 3 & 0^0 \\
6 & 8 & 2 & 0^1 & 1 & 7 & \infty \\
7 & 0^6 & \cancel{0^0} & \cancel{0^0} & \cancel{0^0} & \cancel{\infty} & \cancel{0^0}
\end{array}$$

Рассматриваем 2 случая – **7→1** и **7↛1**, и строим очередной ярус дерева решений:

$7 \not\to 5$, $5 \to 7 \to \mathbf{1}$

$$W_{21} = \begin{array}{c|ccccc}
 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & 0 & 0 & 0 & \infty & 0 \\
2 & \infty & 0 & 0 & 4 & 1 \\
3 & 0 & \infty & 0 & 1 & 0 \\
4 & 0 & 0 & \infty & 3 & 0 \\
6 & 2 & 0 & 1 & 7 & \infty \\
\hline
 & & & & 1 &
\end{array}$$



$7 \not\to 5$, $5 \to 7$, $\mathbf{7 \not\to 1}$

$$W_{22} = \begin{array}{c|cccccc}
 & 1 & 2 & 3 & 4 & 5 & 6 \\
\hline
1 & \infty & 0 & 0 & 0 & 0 & 0 \\
2 & 6 & \infty & 0 & 0 & 4 & 1 \\
3 & 6 & 0 & \infty & 0 & 1 & 0 \\
4 & 7 & 0 & 0 & \infty & 3 & 0 \\
6 & 8 & 2 & 0 & 1 & 7 & \infty \\
7 & \infty & 0 & 0 & 0 & \infty & 0 \\
\hline
 & 6 & & & & &
\end{array}$$

$C_{21}=34+1=35$

$C_{22}=34+6=40$

Т.к. оценки для матриц W_{01} и W_{21} наименьшие, но W_{21} имеет меньшую размерность, чем W_{01}, то переходим к приведенной матрице W_{21}^*:

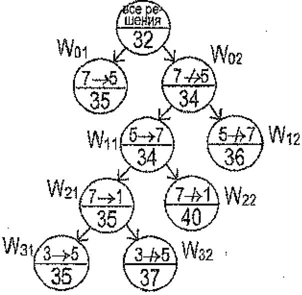
$$W_{21}^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \end{array}\begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0^0 & 0^0 & 0^0 & \infty & 0^0 \\ \infty & 0^0 & 0^0 & 3 & 1 \\ 0^0 & \infty & 0^0 & 0^2 & 0^0 \\ 0^0 & 0^0 & \infty & 2 & 0^0 \\ 2 & 0^1 & 1 & 6 & \infty \end{bmatrix} \end{array}$$

Рассматриваем 2 случая – **3→5** и **3↛5**, и строим очередной ярус дерева решений:

7↛5, **3→5**→7→1

$$W_{31} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ 6 \end{array}\begin{array}{c} \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \begin{bmatrix} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 2 & 0 & 1 & \infty \end{bmatrix} \end{array}$$

$C_{31}=35+0=35$



7↛5, 5→7→1, **3↛5**

$$W_{32} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 6 \\ \\ \end{array}\begin{array}{c} \begin{array}{ccccc} 2 & 3 & 4 & 5 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & 0 & \infty & 0 \\ \infty & 0 & 0 & 3 & 1 \\ 0 & \infty & 0 & \infty & 0 \\ 0 & 0 & \infty & 2 & 0 \\ 2 & 0 & 1 & 6 & \infty \end{bmatrix} \\ \begin{array}{ccccc} & & & 2 & \end{array} \end{array}$$

$C_{32}=35+2=37$

Оценки для W_{01} и W_{31} наименьшие, размерность W_{31} меньше, значит, переходим к W_{31}^*:

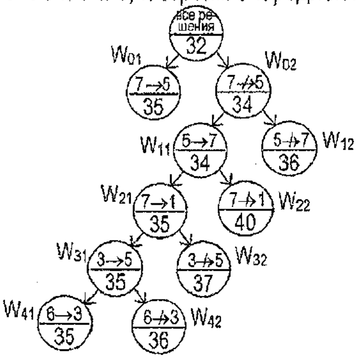
$$W_{31}^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ 6 \end{array}\begin{array}{c} \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \begin{bmatrix} 0^0 & \infty & 0^0 & 0^0 \\ \infty & 0^0 & 0^0 & 1 \\ 0^0 & 0^0 & \infty & 0^0 \\ 2 & 0^1 & 1 & \infty \end{bmatrix} \end{array}$$

Рассматриваем 2 случая – **6→3** и **6↛3**, и строим очередной ярус дерева решений:

7↛5, **6→3**→5→7→1

$$W_{41} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array}\begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{bmatrix} 0 & 0 & \infty \\ \infty & 0 & 1 \\ 0 & \infty & 0 \end{bmatrix} \end{array}$$

$C_{41}=35+0=35$



7↛5, 3→5→7→1,
6↛3

$$W_{42} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ 6 \end{array}\begin{array}{c} \begin{array}{cccc} 2 & 3 & 4 & 6 \end{array} \\ \begin{bmatrix} 0 & \infty & 0 & 0 \\ \infty & 0 & 0 & 1 \\ 0 & 0 & \infty & 0 \\ 2 & \infty & 1 & \infty \end{bmatrix} \end{array}\begin{array}{c} \\ \\ \\ \\ 1 \end{array}$$

$C_{42}=35+1=36$

Оценки для W_{01} и W_{41} наименьшие, размерность W_{41} меньше, значит, переходим к W_{41}^*:

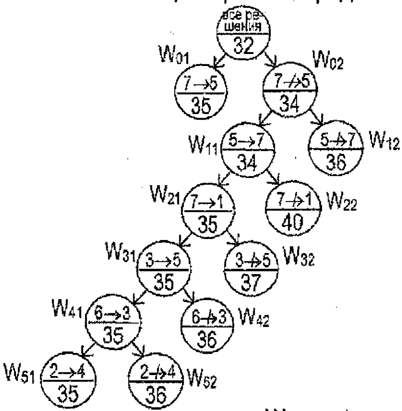
$$W_{41}^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array}\begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{bmatrix} 0^0 & 0^0 & \infty \\ \infty & 0^1 & 1 \\ 0^0 & \infty & 0^1 \end{bmatrix} \end{array}$$

Рассматриваем 2 случая – **2→4** и **$2 \not\to 4$**, и строим очередной ярус дерева решений:

$7 \not\to 5$, $2 \to 4$,
$6 \to 3 \to 5 \to 7 \to 1$

$$W_{51} = \begin{array}{c|cc} & 2 & 6 \\ \hline 1 & 0 & \infty \\ 4 & \infty & 0 \end{array}$$

$C_{51}=35+0=35$



$7 \not\to 5$, $2 \not\to 4$,
$6 \to 3 \to 5 \to 7 \to 1$

$$W_{52} = \begin{array}{c|ccc|c} & 2 & 4 & 6 & \\ \hline 1 & 0 & 0 & \infty & \\ 2 & \infty & \infty & 1 & 1 \\ 4 & 0 & \infty & 0 & \end{array}$$

$C_{52}=35+1=36$

Оценки для W_{01} и W_{41} наименьшие, размерность W_{51} меньше, значит, переходим к W_{51}^*:

$$W_{51}^* = \begin{array}{c|cc} & 2 & 6 \\ \hline 1 & 0 & \infty \\ 4 & \infty & 0 \end{array}$$

К данному моменту получено: 2→4, 6→3→5→7→1. Добавив 2 перехода, разрешенных в W_{51}^* (1→2 и 4→6), получим цикл, содержащий все детали: 2→4→6→3→5→7→1.

Удалив фиктивную деталь из цикла, получаем окончательно искомый порядок обработки:

$2 \to 4 \to 6 \to 3 \to 5 \to \cancel{7} \to 1$

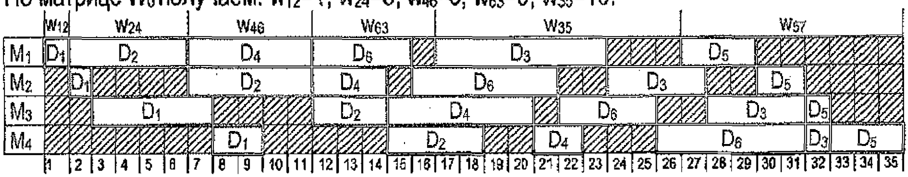
$D_1 \to D_2 \to D_4 \to D_6 \to D_3 \to D_5$ – порядок обработки.

Построим компактное расписание, соответствующее найденному оптимальному порядку обработки. Для этого обратимся к матрицам P и W_0.

$$P = \begin{array}{c|cccccc} & D_1 & D_2 & D_3 & D_4 & D_5 & D_6 \\ \hline M_1 & 1 & 5 & 7 & 5 & 3 & 4 \\ M_2 & 1 & 5 & 4 & 3 & 2 & 6 \\ M_3 & 5 & 3 & 4 & 6 & 1 & 4 \\ M_4 & 2 & 4 & 1 & 2 & 3 & 6 \end{array}$$

$$W_0 = \begin{array}{c|ccccccc} & 1 & 2 & 3 & 4 & 5 & 6 & 7 \\ \hline 1 & \infty & 1 & 1 & 1 & 3 & 1 & 9 \\ 2 & 11 & \infty & 5 & 5 & 11 & 6 & 17 \\ 3 & 13 & 7 & \infty & 7 & 10 & 7 & 16 \\ 4 & 12 & 5 & 5 & \infty & 10 & 5 & 16 \\ 5 & 4 & 3 & 3 & 3 & \infty & 3 & 9 \\ 6 & 13 & 7 & 5 & 6 & 14 & \infty & 20 \\ 7 & 0 & 0 & 0 & 0 & 0 & 0 & \infty \end{array}$$

По матрице P определяем время обработки каждой детали на каждом станке.
По матрице W_0 получаем: $w_{12}=1$, $w_{24}=5$, $w_{46}=5$, $w_{63}=5$, $w_{35}=10$.



Время обработки по полученному расписанию составляет 35 единиц. Задача решена.

2.3. Реализация решения задачи составления расписания горячей обработки (РГО) в Excel

Для решения задачи составления РГО в Excel будет реализована *математическая модель* решения задачи коммивояжера:

<u>*Переменные задачи*</u>:

$x_{ik}, i,k=\overline{1,n}, i \neq k$, – логическая переменная, определяющая наличие дуги (i, k) в цикле, т.е. равна 1, если дуга присутствует в цикле, или равна 0 в противном случае;

$u_i, i=\overline{2,n}$, – параметр связанности вершин в цикле.

<u>*Целевая функция*</u>:

$$z=\sum_{i=1}^{n}\sum_{\substack{k=1\\ i\neq k}}^{n} w_{ik}\cdot x_{ik} \rightarrow \min;$$

- общая стоимость движения коммивояжера по циклу.

<u>*Ограничения задачи*</u>:

$$\begin{cases} x_{ik} \in \{0,1\}, \\ i,k=\overline{1,n}, i\neq k; \end{cases}$$

- (ограничение 1) все переменные равны нулю или единице;

$$\sum_{\substack{k=1\\ i\neq k}}^{n} x_{ik} = 1,\ i=\overline{1,n};$$

- (ограничение 2) из каждой вершины выходит только одна дуга;

$$\sum_{\substack{i=1\\ i\neq k}}^{n} x_{ik} = 1,\ k=\overline{1,n};$$

- (ограничение 3) в каждую вершину входит только одна дуга;

$$\begin{cases} u_i - u_k + (n-1)\cdot x_{ik} \le n-2, \\ \qquad 2 \le i \neq k \le n. \end{cases}$$

- (ограничение 4) все циклы – полные, т.е. состоят из n вершин.

Для решения задачи составления РГО необходимо подготовить данные и формулы в таблице *Excel* и, используя надстройку *Поиск решения*, получить решение.

1. Введем исходные данные (матрицу p, содержащую длительности обработки каждой детали каждым станком), предварительно транспонировав ее.

$$P = \begin{array}{c} \\ M_1 \\ M_2 \\ M_3 \\ M_4 \end{array} \begin{array}{c} \begin{array}{cccccc} D_1 & D_2 & D_3 & D_4 & D_5 & D_6 \end{array} \\ \begin{bmatrix} 1 & 5 & 7 & 5 & 3 & 4 \\ 1 & 5 & 4 & 3 & 2 & 6 \\ 5 & 3 & 4 & 6 & 1 & 4 \\ 2 & 4 & 1 & 2 & 3 & 6 \end{bmatrix} \end{array}$$

Длительности обработки деталей D1, D2, D3, D4, D5, D6 станками M1, M2, M3, M4:

		M1	M2	M3	M4
	D1	1	1	5	2
	D2	5	5	3	4
p=	D3	7	4	4	1
	D4	5	3	6	2
	D5	3	2	1	3
	D6	4	6	4	6

2. Введем матрицу W, предварительно полученную в *Derive*.

	A	B	C	D	E	F	G	H	I	J
10	Время от начала обработки детали Di до начала обработки детали Dj на первом станке:									
11			1	2	3	4	5	6	7	
12		1	100000	1	1	1	3	1	9	
13		2	11	100000	5	5	11	6	17	
14		3	13	7	100000	7	10	7	16	
15	W=	4	12	5	5	100000	10	5	16	
16		5	4	3	3	3	100000	3	9	
17		6	13	7	5	6	14	100000	20	
18		7	0	0	0	0	0	0	100000	

а) на главной диагонали вместо знака ∞ необходимо поставить достаточно большое, по сравнению с исходными данными, число (например, 100 000);

б) элементы последнего столбца (суммарное время обработки каждой детали) рассчитаем формулам:

в ячейку **I12**: **=СУММ(C3:F3)**, которую тиражируем на диапазон **I13:I17**

3. Определим диапазон для искомых значений – матрицы $X=(x_{ik}),\ \ i,k=1,\ldots,n$:

	A	B	C	D	E	F	G	H	I	J
20	Признак того, обрабатывается деталь Dj сразу же после детали Di, или нет:									
21			1	2	3	4	5	6	7	
22		1								0
23		2								0
24		3								0
25	X=	4								0
26		5								0
27		6								0
28		7								0
29			0	0	0	0	0	0	0	

Для этой матрицы должны выполняться требования, которые будем задавать в виде формул на рабочем листе и в виде ограничений в окне *Поиск решения*:

а) в силу ограничения 1 элемент $x_{ik},\ \ i,k=\overline{1,n}$, матрицы X впоследствии окажется равным либо единице (если сразу же за деталью Di обрабатывается деталь Dk), либо нулю (в противном случае);

б) все элементы главной диагонали матрицы X должны равняться нулю, т.к. недопустима ситуация, когда после детали Di обрабатывается эта же деталь Di.

в) формулы, представляющие собой суммы по строкам и суммы по столбцам матрицы X, позволят позже задать ограничения 2 и 3, а именно:

$$\sum_{\substack{k=1\\ i\neq k}}^{n} x_{ik}=1,\ i=\overline{1,n};$$

– после обработки детали Di должна обрабатываться в точности одна из имеющихся деталей, за исключением детали Di (это условие равносильно тому, что сумма элементов в i-ой строке матрицы X должна равняться единице);

в ячейку **J22**: **=СУММ(C22:I22)**, которую тиражируем на ***J23:J28***

$\sum_{\substack{i=1 \\ i \neq k}}^{n} x_{ik} = 1, k = \overline{1,n};$ – перед обработкой детали *Dk* должна обрабатываться в точности одна из имеющихся деталей, за исключением детали *Dk* (это условие равносильно тому, что сумма элементов *k*-ого столбца матрицы *X* должна равняться единице);

в ячейку **C29**: **=СУММ(C22:C28)**, которую тиражируем на **D29:I29**

4. Введем формулу, задающую выражение для целевой функции $z = \sum_{i=1}^{n} \sum_{\substack{k=1 \\ i \neq k}}^{n} w_{ik} \cdot x_{ik}$,

т.е. определяющую общее время обработки всех деталей:

Общее время обработки всех деталей:	0
Количество n всех деталей (включая фиктивную):	7
Сумма всех элементов главной диагонали матрицы X:	0

Для этого вводим в ячейку **J31**: **=СУММПРОИЗВ(C12:I18;C22:I28)**

5. Укажем количество всех деталей, включая фиктивную, т.е. зададим размерность матрицы *W* (в нашем примере в ячейку **J32**: **=7**).

6. Введем формулу, вычисляющую сумму всех элементов главной диагонали матрицы *X* (необходимо для ограничения, контролирующего равенство нулю всех элементов главной диагонали матрицы *X*):

в ячейку **J33**: **=C22+D23+E24+F25+G26+H27+I28**

7. Определим диапазон для параметров связанности – чисел u_i, $i = 2,...,n$:

Числа u_i, i= 2,...,n - параметры связанности задачи:

u2	u3	u4	u5	u6	u7
0	0	0	0	0	0

8. Для каждой пары значений u_i, u_k зададим формулы для вычисления выражений $u_i - u_k + (n-1) \cdot x_{ik}$, которые используются в ограничениях 4:

Ограничения вида: u_i - u_k +(n-1)*x_{ik}<=n-2, где i, k = 2,...,n:

	u2	u3	u4	u5	u6	u7
u2	0	0	0	0	0	0
u3	0	0	0	0	0	0
u4	0	0	0	0	0	0
u5	0	0	0	0	0	0
u6	0	0	0	0	0	0
u7	0	0	0	0	0	0

Например, выражению $u_5 - u_2 + (n-1) \cdot x_{52}$ соответствует формула

=$F$37-C$37+($J$32-1)*D26, содержащаяся в ячейке **C44**.

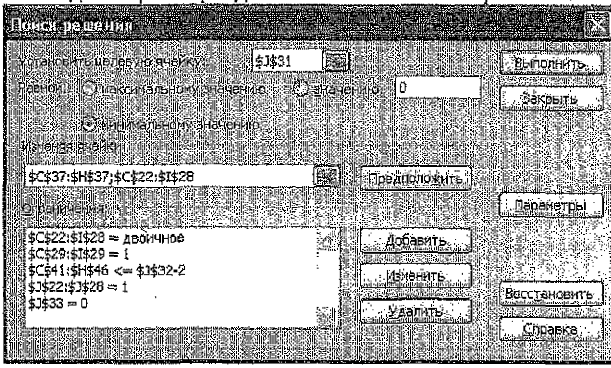
Достаточно заполнить такими формулами первый столбец таблицы, а затем растиражировать на соответствующие диапазоны по строкам.

Окончательно, в формульном виде получим таблицу следующего вида:

Ограничения вида: $u_i - u_k + (n-1)*x_{ik} <= n-2$, где i, k = 2,...,n

	u2	u3	u4	u5	u6	u7
u2	=$C$37-C37+($J$32-1)*D23	=$C$37-D37+($J$32-1)*E23	=$C$37-E37+($J$32-1)*F23	=$C$37-F37+($J$32-1)*G23	=$C$37-G37+($J$32-1)*H23	=$C$37-H37+($J$32-1)*I23
u3	=$D$37-C37+($J$32-1)*D24	=$D$37-D37+($J$32-1)*E24	=$D$37-E37+($J$32-1)*F24	=$D$37-F37+($J$32-1)*G24	=$D$37-G37+($J$32-1)*H24	=$D$37-H37+($J$32-1)*I24
u4	=$E$37-C37+($J$32-1)*D25	=$E$37-D37+($J$32-1)*E25	=$E$37-E37+($J$32-1)*F25	=$E$37-F37+($J$32-1)*G25	=$E$37-G37+($J$32-1)*H25	=$E$37-H37+($J$32-1)*I25
u5	=$F$37-C37+($J$32-1)*D26	=$F$37-D37+($J$32-1)*E26	=$F$37-E37+($J$32-1)*F26	=$F$37-F37+($J$32-1)*G26	=$F$37-G37+($J$32-1)*H26	=$F$37-H37+($J$32-1)*I26
u6	=$G$37-C37+($J$32-1)*D27	=$G$37-D37+($J$32-1)*E27	=$G$37-E37+($J$32-1)*F27	=$G$37-F37+($J$32-1)*G27	=$G$37-G37+($J$32-1)*H27	=$G$37-H37+($J$32-1)*I27
u7	=$H$37-C37+($J$32-1)*D28	=$H$37-D37+($J$32-1)*E28	=$H$37-E37+($J$32-1)*F28	=$H$37-F37+($J$32-1)*G28	=$H$37-G37+($J$32-1)*H28	=$H$37-H37+($J$32-1)*I28

9. Введем параметры диалогового окна *Поиск решения*:



– укажем целевую ячейку, содержащую общее время обработки всех деталей;
– установим ее равной минимальному значению;
– зададим изменяемые ячейки (диапазон с числами u_i и диапазон со значениями матрицы X),
– добавим ограничения:
  - значениями элементов матрицы X могут быть только двоичные числа 0 и 1,
  - сумма значений в каждом столбце и в каждой строке матрицы X равна 1,
  - сумма элементов главной диагонали матрицы X равна 0,
  - все значения, рассчитанные по формулам $u_i - u_k + (n-1) \cdot x_{ik}$, не должны превышать n-2.

В Параметрах окна *Поиска решения* установим *Линейную модель.*

10. В результате выполнения *Поиска решения* окажутся заполненными:

а) таблица со значениями u_i, но они являются вспомогательными и к ответу на вопрос задачи отношения не имеют;

б) таблица, в которой будут найдены значения матрицы X, а также будет найдено значение целевой функции – минимальное время, за которое можно обработать при заданных условиях все детали (в нашем примере – 35 единиц).

По единичным значениям матрицы X определим порядок, в котором должны обрабатываться детали, чтобы общее время их обработки было минимальным. Формирование решения начнем с фиктивной детали. Видим, что $x_{71} = 1$, значит, сразу после детали *D7* должна обрабатываться деталь *D1*. Аналогично, сразу после детали *D1* должна обрабатываться деталь *D2*, после *D2* – *D4*, после *D4* – *D6*, после *D6* – *D3*, после *D3* – *D5*, после *D5* – *D7* . Таким образом, получен искомый цикл:

$$7 - 1 - 2 - 4 - 6 - 3 - 5 - 7.$$

Удалив из него фиктивную 7-ю деталь, получим следующий порядок обработки:
D1 – D2 – D4 – D6 – D3– D5.

Признак того, обрабатывается деталь Dj сразу же после детали Di, или нет:									
		1	2	3	4	5	6	7	
	1	0	1	0	0	1,09187E-11	0	0	1
	2	0	0	3,41995E-16	1	0	0	0	1
	3	0	0	0	0	1	1,09184E-11	6,25E-16	1
X=	4	0	0	1,09182E-11	0	0	1	0	1
	5	0	0	0	0	0	0	1	1
	6	0	1,09182E-11	1	0	0	0	0	1
	7	1	0	0	0	2,91406E-16	0	0	1
		1	1	1	1	1	1	1	
					Общее время обработки всех деталей:				35
				Количество n всех деталей (включая фиктивную):					7
			Сумма всех элементов главной диагонали матрицы X:						0

Общее время обработки (35 единиц) совпало с тем, которое было получено при решении данной задачи вручную.

Вспомогательный файл **Контрольные задания по ДМ.xls** для решения задач контрольной работы находится в локальной вычислительной сети БрГТУ в папке:

U:\ VT&PM \ Zaoch_f \ Дискр_мат

3.1. Сетевая модель технологического процесса

Задача. Для достижения заданных чертежом размеров и технических требований все поверхности детали проходят несколько стадий обработки, преобразующих ее из состояния заготовки в состояние готовой поверхности. Предполагается, что поверхность заготовки можно обработать не единственным способом (например, разными видами оборудования). Требуется так распределить заготовки по доступному оборудованию, чтобы число заготовок, обрабатываемых в единицу времени, было максимальным.

В этом случае задачу можно представить в виде сети. Вершины сети – это различные состояния поверхности, *начиная от заготовки (состояние 1) и заканчивая готовой поверхностью* (состояние 8). Источник – это заготовка в исходном (необработанном) состоянии, а сток – это готовая поверхность. Дуга сети обозначает оборудование, с помощью которого поверхность переходит из одного промежуточного состояния в другое. Для каждого доступного вида оборудования, которое переводит заготовку из состояния i в состояние j, известна величина c_{ij}. c_{ij} - это число заготовок, которое может быть обработано в единицу времени (т.е. пропускная способность соответствующей дуги).

Потоки в сетях.

Транспортной сетью называется орграф G=<V, R> с множеством вершин $V=\{v_1,v_2,\ldots,v_n\}$, для которого выполняются условия:

1) существует одна и только одна вершина v_1, называемая источником, такая, что ни одна дуга не заходит в v_1;

2) существует одна и только одна вершина v_n, называемая стоком, такая, что из v_n не исходит ни одной дуги;

3) каждой дуге $r\in R$ поставлено в соответствие целое число $c(r)\geq 0$ – пропускная способность дуги.

Вершины в сети, отличные от источника и стока, называется промежуточными.

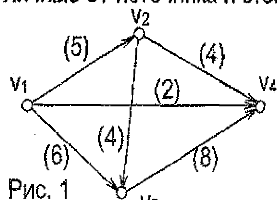


Рис. 1

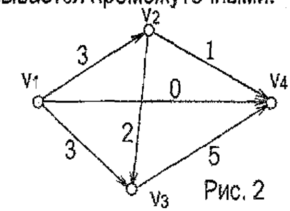


Рис. 2

На рис. 1 показан пример транспортной сети. Здесь v_1 – источник, v_4 – сток, v_2 и v_3 – промежуточные вершины. При каждой дуге в скобках указана ее пропускная способность. Например, $c((v_1,v_4)) = 2$.

Функция φ, определенная на множестве R дуг транспортной сети G и принимающая целочисленные значения, называется допустимым потоком в транспортной сети G, если:

1) для любой дуги $r\in R$ величина $\varphi(r)$ (поток по дуге r), удовлетворяет условию: $0\leq\varphi(r)\leq c(r)$;

2) для любой промежуточной вершины v выполняется равенство:

сумма потоков по дугам, заходящим в v, равна сумме потоков по дугам, исходящей из v. Говоря иными словами, допустимый поток, пущенный в сети, – это приписанные дугам этой сети целые неотрицательные числа, каждое из которых не превосходит пропускной способности соответствующей дуги, а сумма потоков по дугам, заходящим в любую промежуточную вершину, равна сумме потоков по дугам, исходящим из нее.

Величиной потока φ в транспортной сети G называется величина $\overline{\varphi}$, равная сумме потоков по всем дугам, заходящим в v_n, или, что то же самое, – величина, равная сумме потоков по всем дугам, исходящим из v_1.

Пусть φ - поток в транспортной сети G. Дуга $r\in R$ в G называется насыщенной, если поток по ней равен ее пропускной способности, то есть если $\varphi(r)=c(r)$.

Поток φ называется полным, если любой путь в G из v_1 в v_n содержит по крайней мере одну насыщенную дугу.

Поток φ называется максимальным, если его величина $\overline{\varphi}$ принимает максимальное значение по сравнению с другими допустимыми потоками в транспортной сети G.

На рис. 2 показан один из возможных допустимых потоков в заданной транспортной сети. Например, в промежуточную вершину v_2 по единственной входящей в нее дуге заходит 3 единицы потока, а по двум исходящим из нее дугам выходит также 3 единицы потока. Ни одна из дуг здесь не является насыщенной. Следовательно, данный поток не является полным. Его величина равна 3+0+3=1+0+5=6, т.е. $\overline{\varphi}=6$.

Максимальный поток φ обязательно является полным (т.к. иначе в G существует некоторый простой путь из v_1 в v_n, не содержащий насыщенных дуг, а значит, можно увеличить на 1 потоки по всем дугам из этого пути и тем самым увеличить на единицу $\overline{\varphi}$, что противоречит условию максимальности потока). Но существуют полные потоки, не являющиеся максимальными. Полный поток можно рассматривать как приближение к максимальному.

Алгоритм построения полного потока в транспортной сети.

Шаг 1. Полагаем $\forall r \in R$ $\varphi(r)=0$ (пускаем нулевой поток по всем дугам). Полагаем G'=G.

Шаг 2. Удаляем из орграфа G' все дуги, являющиеся насыщенными при потоке φ в транспортной сети G. Полученный орграф снова обозначим через G'.

Шаг 3. Ищем в G простую цепь ω из v_1 в v_n. Если такой цепи нет, то φ - искомый полный поток в транспортной сети G. В противном случае переходим к шагу 4.

Шаг 4. Увеличиваем поток $\varphi(r)$ по каждой дуге v из ω на одинаковую величину $a>0$ такую, что, по крайней мере, одна дуга из ω оказывается насыщенной, а потоки по остальным дугам из ω не превышают их пропускных способностей. При этом величина потока $\overline{\varphi}$ также увеличивается на a, а сам поток φ в транспортной сети G остается допустимым (поскольку в каждую промежуточную вершину, содержащуюся в ω, дополнительно вошло a единиц потока и из нее вышло также a единиц потока). После этого переходим к шагу 2.

Алгоритм нахождения максимального потока в сети.

1. На исходном графе пускаем нулевой поток или любой из найденных полных потоков.

2. Строим граф (граф приращений), совпадающий с исходным, каждой дуге которого приписываем пару, первая компонента которой равна разности пропускной способности этой дуги и потока, пущенного по этой дуге, а вторая – потоку, пущенному по этой дуге. Первая компонента показывает, на какую величину можно увеличить поток по этой дуге, а вторая – на какую величину можно этот поток уменьшить.

3. Находим на графе приращений путь из источника в сток, такой, на котором возможно увеличение потока. При этом разрешается движение по дугам в направлении, противоположном заданному, т.е. при необходимости обратные дуги добавляются искусственно. По найденному пути пропускаем максимально возможный поток. При этом на прямых дугах поток увеличиваем, а на обратных - уменьшаем.

4. Если такого пути нет, то указанный на исходном графе поток является максимальным. Если такой путь есть, то пускаем найденный увеличенный поток по исходному графу и переходим к пункту 2.

Для заданной транспортной сети G и допустимого потока φ в этой сети орграф приращений имеет те же вершины, что и сеть G. Каждой дуге $r=(v,w) \in R$ сети D в орграфе приращений соответствуют две дуги: r и $r'=(w,v)$ - дуга, противоположная по направлению дуге r.

Пример 1. Построение полного потока в заданной транспортной сети G=<V,R> (рис.3).

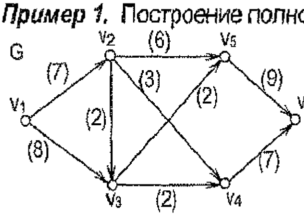


Рис. 3.

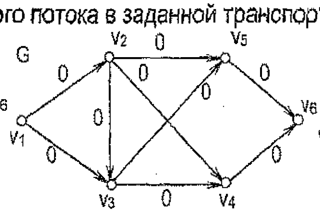


Рис. 4.

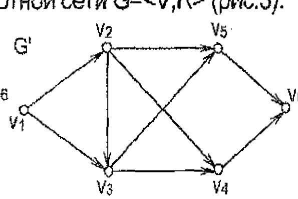


Рис. 5.

1. Начнем с нулевого потока (рис. 4). Полагаем G'=G. При нулевом потоке отсутствуют насыщенные дуги.

2. Выделим в G' простой путь $\omega_1=v_1v_2v_4v_6$ и увеличим потоки по дугам из ω_1 на 3 (до насыщения дуги (v_2, v_4)) (рис. 6). В результате получим поток $\varphi=\varphi_1$, содержащий одну насыщенную дугу, и на графе G пометим насыщенную дугу знаком × (аналогично будем помечать все насыщенные дуги) (рис. 7). Удалим эту дугу из G', оставшийся орграф снова обозначим через G' (рис. 8).

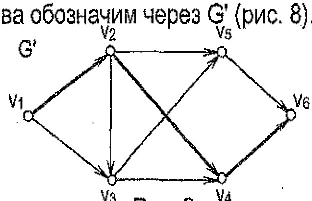


Рис. 6.

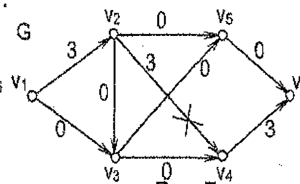


Рис. 7.

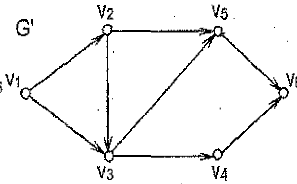


Рис. 8.

3. Выделим в G' простой путь $\omega_2=v_1v_2v_3v_4v_6$ и увеличим потоки по дугам из ω_2 на 2 (до насыщения дуг (v_2,v_3),(v_3,v_4)) (рис.9). В результате получим поток $\varphi=\varphi_2$, содержащий 3 насыщенные дуги (рис. 10). Удалим вновь полученные насыщенные дуги из G', оставшийся орграф снова обозначим G' (рис. 11).

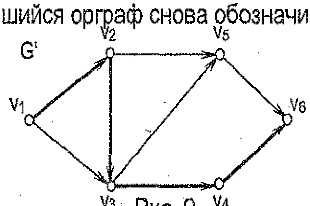


Рис. 9.

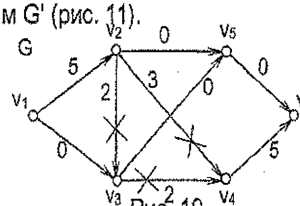


Рис. 10.

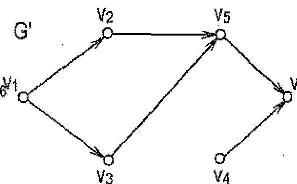


Рис. 11.

4. Выделим в G' простой путь $\omega_3=v_1v_3v_5v_6$ и увеличим потоки по дугам ω_3 на 2 (до насыщения дуги(v_3, v_5)) (рис. 12). В результате получим поток $\varphi=\varphi_3$, содержащий 4 насыщенные дуги (рис. 13). Удалим вновь полученную насыщенную дугу из G', оставшийся орграф снова обозначим через G' (рис. 14).

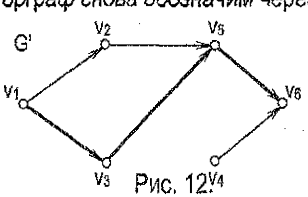


Рис. 12.

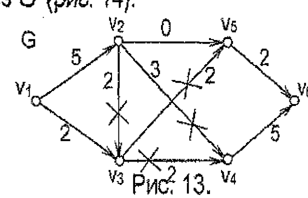


Рис. 13.

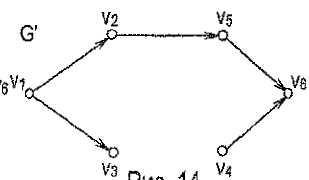


Рис. 14.

5. Выделим в G' простой путь $\omega_4=v_1v_2v_5v_6$ и увеличим потоки по дугам из ω_4 на 2 (до насыщения дуги (v_1v_2)) (рис. 15). В результате получим поток $\varphi=\varphi_4$, содержащий 5 насыщенных дуг (рис. 16). Удалим вновь полученную насыщенную дугу из G', оставшийся орграф снова обозначим через G' (рис. 17).

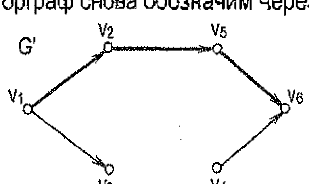


Рис. 15.

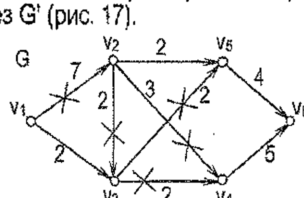


Рис. 16.

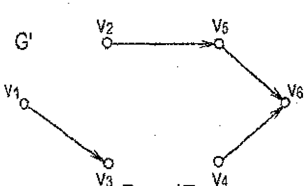


Рис. 17.

6. Заметим, что в G' не существует пути из v_1 в v_6, следовательно, в транспортной сети G с потоком φ_4 не существует пути из v_1 в v_6, который не содержал бы насыщенных дуг, то есть поток φ_4 является полным. Величина $\overline{\varphi_4}$ полученного полного потока равна 9 (7+2 = 4+5).

Пример 2. Нахождение максимального потока по уже полученному полному потоку.

1. Построим орграф приращений для полного потока φ_4 (рис. 18).

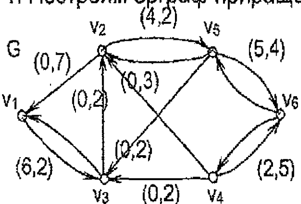


Рис. 18.

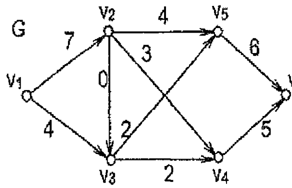


Рис. 19.

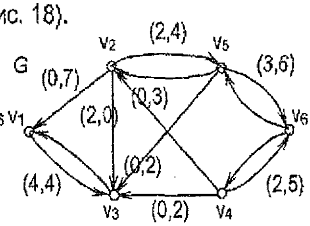


Рис. 20.

2. Попытаемся на этом орграфе найти путь из источника в сток (такой, на котором возможно увеличение потока). В этом орграфе имеется простой путь $v_1v_3v_2v_5v_6$, по которому возможно изменение потока (увеличение по дуге (v_1,v_3), уменьшение по дуге (v_3,v_2), увеличение по дугам (v_2,v_5) и (v_5,v_6)).

Изменяем поток вдоль данного пути на максимально допустимую величину (на 2, до обнуления потока по дуге (v_2, v_3)). В результате получим поток φ_5 (рис. 19).

Строим граф приращений для этого потока (рис. 20). В полученном графе приращений не существует пути из v_1 в v_6, для возможного изменения потока, значит, поток φ_5 является максимальным, и при этом $\overline{\varphi_5}=11$.

3.2. Пример оформления решения задачи о максимальном потоке в сети

	1	2	3	4	5	6
1	-	7	8	-	-	-
2	-	-	2	3	6	-
3	-	-	-	2	2	-
4	-	-	-	-	-	7
5	-	-	-	-	-	9

По исходным данным строим транспортную сеть с источником в вершине v_1 и стоком в вершине v_6, при каждой дуге в скобках указываем ее пропускную способность.

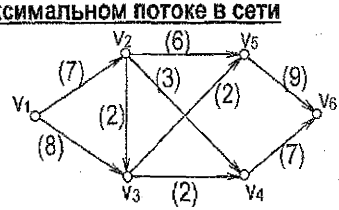


Рис 1. Транспортная сеть.

1. Ищем пути из v_1 в v_6 на исходном графе и увеличиваем поток по дугам каждого из них до тех пор, пока это возможно:

$\omega_1=v_1v_2v_4v_6$, a=3,

$\omega_2=v_1v_2v_3v_4v_6$, a=2,

$\omega_3=v_1v_3v_5v_6$, a=2,

$\omega_4=v_1v_2v_5v_6$, a=2.

При этом дугам приписываем пары чисел (x,y):

x – сколько еще можно пустить по дуге,

y – сколько уже пущено.

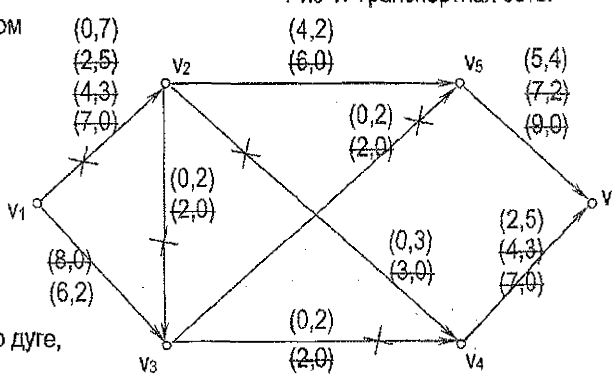


Рис.2. Отыскание полного потока.

2. Пускаем в сети найденный полный поток.
Для этого каждой дуге ставим в соответствие число, окончательно полученное в пункте 1 при отыскании полного потока, второе в каждой паре (см. рис.2).
Величина полученного полного потока равна 9:

$\bar{\varphi}_{полн} = 3+2+2+2 = 7+2 = 4+5 = 9.$

Она может быть найдена одним из способов:

- 3+2+2+2 – все, что пускали по путям из v_1 в v_6;
- 7+2 – все, что вышло из v_1 (из источника);
- 4+5 – все, что зашло в v_6 (в сток).

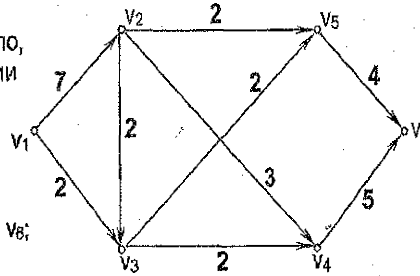


Рис.3. Найденный полный поток.

3. Строим граф приращений (ГП) по найденному полному потоку. ГП строится на тех же вершинах, что и исходный. Дуги графа приращений строим так:
- если дуге на рис.2 окончательно приписана пара вида (0,x), где x≠0, то в ГП меняем направление этой дуги на противоположное;
- если дуге на рис.2 окончательно приписана пара вида (x,0), где x≠0, то в ГП строим такую же дугу, не меняя ее направления;
- если дуге на рис.2 окончательно приписана пара вида (x,y), где x≠0, y≠0, то в ГП вместо этой дуги строим две противоположно направленные дуги.

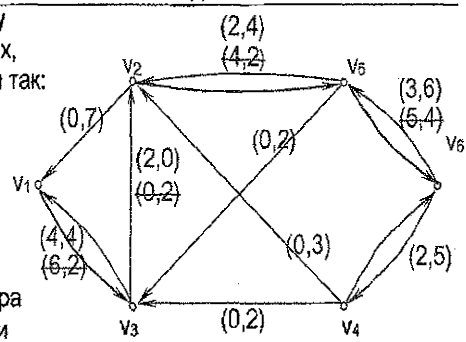


Рис.4. Граф приращений.

Ищем путь из v_1 в v_6 на графе приращений. Такой путь существует: $\omega_5 = v_1v_3v_2v_5v_6$, a=2.
Если в найденном пути направление дуги совпадает с исходным, то по этой дуге поток увеличиваем, если направление дуги противоположно исходному, то по этой дуге поток уменьшаем.
Поток изменился (по дугам v_1v_3, v_2v_5, v_5v_6 увеличился, а по дуге v_3v_2 уменьшился на 2 единицы).

4. Пускаем в сети новый полный поток и строим соответствующий ему новый граф приращений:

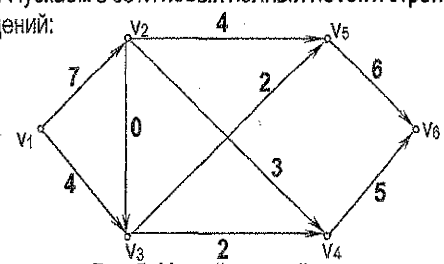


Рис.5. Новый полный поток.

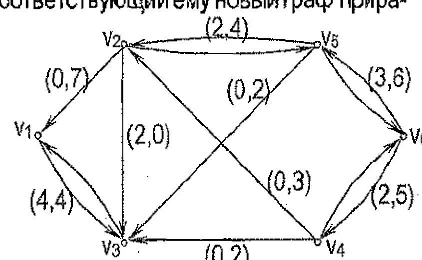


Рис.6. Новый граф приращений.

Дальнейшее увеличение потока невозможно – в новом графе приращений нет путей из v_1 в v_6: из источника v_1 можно выйти только в вершину v_3, а из v_3 ни в какую другую вершину движение невозможно, т.к. из v_3 не исходит ни одна дуга.

5. Если увеличение полного потока невозможно, значит, этот полный поток является максимальным. Таким образом, найден максимальный поток в сети (этот поток изображен на рис.5), его величина: $\bar{\varphi}_{max} = 3+2+2+2+2 = 7+4 = 6+5 = 11.$

3.3. Решение задачи о максимальном потоке в Excel

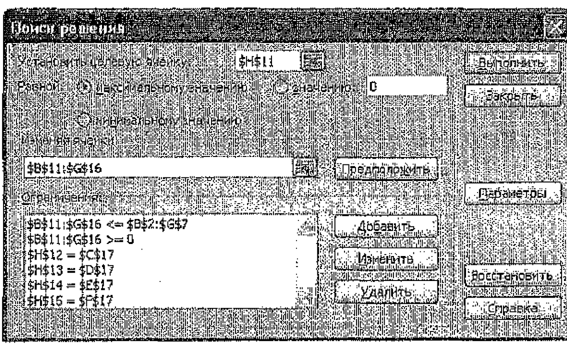
	A	B	C	D	E	F	G	H
1		1	2	3	4	5	6	
2	1	0	7	8	0	0	0	
3	2	0	0	2	3	6	0	
4	3	0	0	0	2	2	0	
5	4	0	0	0	0	0	7	
6	5	0	0	0	0	0	9	
7	6	0	0	0	0	0	0	
8								
9								
10		1	2	3	4	5	6	
11	1	0	0	0	0	0	0	0
12	2	0	0	0	0	0	0	0
13	3	0	0	0	0	0	0	0
14	4	0	0	0	0	0	0	0
15	5	0	0	0	0	0	0	0
16	6	0	0	0	0	0	0	
17			0	0	0	0	0	

В диапазон B2:G7 вводим пропускные способности дуг.

Диапазон D11:G16 оставляем пустым – туда будет выведен поток, найденный Excel с помощью надстройки Поиск решения.

В ячейку H11 вводим формулу: =СУММ(B11:G11). Эта сумма показывает, сколько единиц потока выходит из вершины 1 (из источника). Тиражируем эту формулу на диапазон H12:H15 (эти суммы показывают, сколько единиц потока выходит из вершин со 2-й по 5-ю соответственно).

В ячейку G17 вводим формулу: =СУММ(G11:G16). Эта сумма показывает, сколько единиц потока заходит в вершину 6 (в сток). Тиражируем эту формулу на диапазон C17:F17 (эти суммы показывают, сколько единиц потока заходят в вершины со 2-й по 5-ю соответственно).



Устанавливаем целевую ячейку H11, значение в которой показывает, сколько единиц потока выходит из источника, равной максимальному значению.

Целевой ячейкой может быть и ячейка G17 (сколько единиц потока заходит в сток).

Изменяемые ячейки: D11:G16, в них разместится найденный поток. Задаем ограничения:

$B$11:$G$16 <= $B$2:$G$16 – для каждой дуги пущенный по ней поток не должен превышать пропускной способности этой дуги;

$B$11:$G$16 >= 0 – поток по каждой дуге должен быть неотрицательным числом;

$H$12 = $C$17 – сколько единиц потока зашло во 2-ю вершину, столько же из нее должно и выйти. Аналогично строятся оставшиеся три ограничения.

В параметрах окна *Поиска решения* указываем *линейную модель.*

После выполнения поиска решения в диапазоне B11:G16 получим пущенный по дугам сети максимальный поток, величина которого равна 11 (значения в ячейках H11 и G17). Найденный в Excel поток может отличаться от того, который найден вручную, но величины этих потоков должны совпадать.

	A	B	C	D	E	F	G	H
10		1	2	3	4	5	6	
11	1	0	7	4	0	0	0	11
12	2	0	0	0	3	4	0	7
13	3	0	0	0	2	2	0	4
14	4	0	0	0	0	0	5	5
15	5	0	0	0	0	0	6	6
16	6	0	0	0	0	0	0	
17			7	4	5	6	11	

Литература

1. Кузнецов О.П., Адельсон-Вельский Г.М. Дискретная математика для инженера. – М.: Энергия, 1987.
2. Капустин Н.М., Дьяконова Н.П., Кузнецов П.М. Автоматизация машиностроения. – М.: Высшая школа, 2003. – 223с.
3. Нефедов В.Н., Осипова В.А. Курс дискретной математики. – М.: Издательство МАИ, 1992. – 262с.
4. Рыбников К.А. Введение в комбинаторный анализ. М., 1972.
5. Тимковский В.Г. Дискретная математика в мире станков и деталей. Введение в математическое моделирование задач дискретного производства. – М.: Наука, 1992. – 144с.

СОДЕРЖАНИЕ

УЧЕБНОЕ ИЗДАНИЕ

Составители:
Ирина Владимировна Тузик
Татьяна Георгиевна Хомицкая
Людмила Константиновна Рамская

МЕТОДИЧЕСКИЕ УКАЗАНИЯ

по выполнению контрольной работы
по дисциплине **«Дискретная математика»**
для студентов специальности
1 – 36 01 01 «Технология машиностроения»
заочной формы обучения

Ответственный за выпуск: Тузик И.В.
Редактор: Боровикова Е.А.
Компьютерная верстка: Боровикова Е.А.
Корректор: Никитчик Е.В.

Подписано к печати 11.11.2013 г. Формат 60x84 $^{1}/_{16}$. Бумага «Снегурочка».
Гарнитура Arial. Усл. печ. л. 2,55. Уч. изд. л. 2,75. Заказ № 1207. Тираж 100 экз.
Отпечатано на ризографе Учреждения образования
«Брестский государственный технический университет»
224017, г. Брест, ул. Московская, 267.