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 2016

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

1.1. Задача о наилучшей загрузке станка

<u>*Задача.*</u> Станок налажен на обработку деталей определенной группы. Выделен фонд времени T непрерывной работы станка с данной наладкой. Известно число n деталей в группе, длительность обработки t_i и ценность c_i каждой детали в группе (i=1,...n). Предполагается, что всю группу нельзя обработать в течение выделенного фонда времени, длительности обработки деталей могут быть разными, очередность обработки деталей не важна. Нужно отобрать из группы детали для загрузки станка в течение выделенного фонда времени так, чтобы общая ценность отобранных деталей была максимальной.

<u>***Задания для выполнения.***</u>

1. Составить и записать математическую модель поставленной задачи в соответствии с данными своего варианта.
2. Найти <u>все</u> решения данной задачи методом ветвей и границ.
3. Решить поставленную задачу в Excel с помощью надстройки «Поиск решения», сформировав *Отчет по результатам.*

<u>Решение задачи (пример).</u> Запишем математическую модель задачи:

$$5\cdot x_1 + 4\cdot x_2 + 2\cdot x_3 + 3\cdot x_4 + 6\cdot x_5 \to max$$

$$30\cdot x_1 + 40\cdot x_2 + 20\cdot x_3 + 20\cdot x_4 + 30\cdot x_5 \le 70,\ x_i \in \{0,1\},\ i=1,\dots,5.$$

Найдем удельную ценность каждой детали (ценность на единицу времени). В таблице слева в последнем столбце найден порядок деталей в группе в соответствии с убыванием их удельных ценностей. Разместим детали в полученном порядке (в таблице справа).

i	c_i	t_i	c_i/t_i	Порядок
1	5	30	0,167	②
2	4	40	0,1	④
3	2	20	0,1	⑤
4	3	20	0,15	③
5	6	30	0,2	①

<u>*Примечание.*</u> Если несколько деталей имеет одинаковую удельную ценность (в примере – 2-я и 3-я), то они считаются равноценными и между собой могут быть расположены в произвольном порядке.

i	c_i	t_i	c_i/t_i	
5	6	30	0,2	①
1	5	30	0,167	②
4	3	20	0,15	③
2	4	40	0,1	④
3	2	20	0,1	⑤

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 обработки при выборе очередной детали не превысит заданного (70). В нашем случае 5-я и 1-я детали выбираются целиком, поэтому $x_5=x_1=1$. Их обработка займет $30\cdot x_5+30\cdot x_1=60$ единиц времени, а следующая, 4-я деталь, требует для своей обработки 20 единиц времени и не может быть целиком обработана за оставшиеся 70-60=10 единиц времени. Поэтому берем только часть 4-й детали, такую, которая может быть обработана в точности за оставшиеся 10 единиц времени, т.е. $x_4 = 1/2$. При этом последние в списке 2-я и 3-я детали вообще не могут быть выбраны, т.к. выделенный фонд времени исчерпан, поэтому $x_2=x_3=0$. Тогда общее время обработки выбранных деталей составит $30\cdot x_5+30\cdot x_1+20\cdot x_4+40\cdot x_2+20\cdot x_3=30\cdot 1+30\cdot 1+20\cdot 1/2+40\cdot 0+20\cdot 0=70$, а их суммарная ценность – $6\cdot x_5+5\cdot x_1+3\cdot x_4+4\cdot x_2+2\cdot x_3= 6\cdot 1+5\cdot 1+3\cdot 1/2+4\cdot 0+2\cdot 0=12\ 1/2$. Т.е. верхняя оценка общей ценности равна 12 1/2. Эта величина является именно оценкой, т.к. при ее вычислении использовалось дробное значение x_i (i=4). В процессе решения задачи эта величина может только уменьшиться (в лучшем случае, суммарная ценность окажется равной 12). 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	6	30	1	30	6
1	5	30	1	60	11
4	3	20	1/2	70	12 1/2
2	4	40	0	70	12 1/2
3	2	20	0	70	12 1/2

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 деталям, для которых в процессе решения будут получаться дробные значения x_i. Например, в таблице №0 дробное значение получено для детали номер 4 (x_4=1/2). Поэтому далее будем рассматривать 2 случая: x_4 = 1 или x_4 = 0, т.е. либо брать в обработку 4-ю деталь, либо нет.

Процесс решения будем сопровождать построением дерева решений. В корневой вершине дерева указываем верхнюю оценку общей ценности, полученную в таблице с номером 0 (у нас это 12 1/2). 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,...,5). 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,...,5;
– в этой вершине полученная оценка меньше, чем уже найденное значение целевой функции при целочисленных значениях x_i , i=1,...,5.

Построение таблиц

В таблице 0 дробное значение принимает x_4 = 1/2. Значит, выполняем ветвление по детали номер 4. Этому ветвлению будет соответствовать новая таблица №1. Ее получаем так:

- Исходные данные (столбцы i, c_i, t_i) будем брать из предыдущей таблицы №0.
- Строку, соответствующую 4-й детали, запишем в качестве 1-й строки новой таблицы.
- Остальные строки таблицы 0 сместим на одну вниз, не изменяя их порядок.
- Добавляем столбцы x_i, $\Sigma(t_i \cdot x_i)$, $\Sigma(c_i \cdot x_i)$ для x_4 = 1 и такие же столбцы для x_4 = 0.
- Для x_4 = 1 заполняем сначала столбцы x_i и $\Sigma(t_i \cdot x_i)$:

  полагаем x_4 = 1 (это обязательное условие), тогда $\Sigma(t_i \cdot x_i)$=20·1=20, из 70 единиц выделенного фонда времени остаются свободными 50 единиц;

  деталь 5 требует 30 единиц времени для обработки, на данный момент имеется 50 свободных единиц времени, значит, полагаем x_5 = 1, тогда $\Sigma(t_i \cdot x_i)$=20·1+30·1=50, из 70 остаются свободными 20 единиц времени;

  деталь 1 требует 30 единиц времени для обработки, а на данный момент имеется только 20 свободных единиц времени – меньше, чем нужно. В этом случае полагаем $x_1 = \frac{\text{количество свободного времени}}{\text{количество времени на обработку}} = \frac{20}{30} = 2/3$, тогда $\Sigma(t_i \cdot x_i)$=20·1+30·1+30·2/3=70, выделенный фонд времени исчерпан, значит, 2-ю и 3-ю детали не берем: x_2 = 0, x_3 = 0.

  По найденным x_i заполняем столбец $\Sigma(c_i \cdot x_i)$ и получаем оценку суммарной ценности взятых деталей при x_4 = 1: $\Sigma(c_i \cdot x_i)$ = 3·1+6·1+5·2/3+4·0+2·0=12 1/3.

Для $x_4 = 0$ заполняем сначала столбцы x_i и $\Sigma(t_i \cdot x_i)$:

полагаем $x_4 = 0$ (это обязательное условие), тогда $\Sigma(t_i \cdot x_i)=20\cdot0=0$, из 70 единиц выделенного фонда времени все 70 остаются свободными;

деталь 5 требует 30 единиц времени для обработки, на данный момент имеется 70 свободных единиц времени, значит, полагаем $x_5 = 1$, тогда $\Sigma(t_i \cdot x_i)=20\cdot0+30\cdot1=30$, из 70 остаются свободными 40 единиц;

деталь 1 требует 30 единиц времени для обработки, на данный момент имеется 40 свободных единиц времени, значит, полагаем $x_1 = 1$, тогда $\Sigma(t_i \cdot x_i)=20\cdot0+30\cdot1+30\cdot1=60$, из 70 остаются свободными 10 единиц;

деталь 2 требует 40 единиц времени для обработки, на данный момент имеется 10 свободных единиц; полагаем $x_2=10/40=1/4$, тогда $\Sigma(t_i \cdot x_i)=20\cdot0+30\cdot1+30\cdot1+40\cdot1/4=70$, фонд времени исчерпан, значит, $x_3 = 0$.

По найденным x_i заполняем столбец $\Sigma(c_i \cdot x_i)$ и получаем оценку суммарной ценности взятых деталей при $x_4 = 0$: $\Sigma(c_i \cdot x_i) = 3\cdot0+6\cdot1+5\cdot1+4\cdot1/4+2\cdot0=12$.

И при $x_4 = 1$, и при $x_4 = 0$ получены не точные значения суммарной ценности, а ее <u>оценки</u>, т.к. в обоих случаях при их вычислении использовались дробные значения x_i.

Строим соответствующее ветвление на дереве, а затем из всех открытых в данный момент вершин выбираем ту, у которой оценка больше (в данном случае это вершина с оценкой $12\ ^1/_3$).

Значит, далее будем рассматривать таблицу 1 при $x_4 = 1$ (т.е. таблицу, где получена оценка, соответствующая выбранной вершине). В этой таблице дробное значение принимает $x_1 = 2/3$, значит, ветвление будем вести по 1-й детали. Этому ветвлению будет соответствовать новая таблица (с номером 2). Ее строим аналогично предыдущей таблице, но для нее нужно учитывать новое обязательное условие: $x_4 = 1$ (и для $x_1 = 1$, и для $x_1 = 0$).

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, расположенные выше по дереву по отношению к рассматриваемой вершине (по которой ведется ветвление). Значения x_i, которые являются обязательными условиями ветвления и не могут быть изменены при заполнении таблиц, выделены в них жирным шрифтом.

В каждой таблице, начиная с первой, указано, из какой предыдущей таблицы она была получена.

В завершении по таблицам, содержащим решения (у нас это таблицы №5 и №8), записываем ответ к поставленной задаче. В ответе должен быть указан каждый набор деталей, имеющий максимальную суммарную ценность, а также общее время обработки всех деталей, входящих в набор.

<u>*Примечание.*</u> Расчетные таблицы могут быть построены вручную либо с использованием MS Excel.

Математическая модель:

$$5\cdot x_1+4\cdot x_2+2\cdot x_3+3\cdot x_4+6\cdot x_5 \to \max$$

$$30\cdot x_1+40\cdot x_2+20\cdot x_3+20\cdot x_4+30\cdot x_5 \le 70$$

$$x_i \in \{0,1\},\ i=1,\dots,5.$$

i	c_i	t_i	c_i/t_i	
1	5	30	0,167	②
2	4	40	0,1	④
3	2	20	0,1	⑤
4	3	20	0,15	③
5	6	30	0,2	①

0)	i	c_i	t_i	x_i	$\Sigma(t_i\cdot x_i)$	$\Sigma(c_i\cdot x_i)$
	5	6	30	1	30	6
	1	5	30	1	60	11
	4	3	20	1/2	70	12 1/2
	2	4	40	0	70	12 1/2
	3	2	20	0	70	12 1/2

1)								
из т.0)			$x_4=1$			$x_4=0$		
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	20	1	20	3	0	0	0
5	6	30	1	50	9	1	30	6
1	5	30	2/3	70	12 1/3	1	60	11
2	4	40	0	70	12 1/3	1/4	70	12
3	2	20	0	70	12 1/3	0	70	12

2)			$x_4=1$					
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	5	30	1	30	5	0	0	0
4	3	20	1	50	8	1	20	3
5	6	30	2/3	70	12	1	50	9
2	4	40	0	70	12	1/2	70	11
3	2	20	0	70	12	1	70	11

3)			$x_4=1,\ x_1=1$					
из т.2)			$x_5=1$			$x_5=0$		
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	6	30	1	30		0	0	0
1	5	30	1	60		1	30	5
4	3	20	1	80		1	50	8
2	4	40	нет решений			1/2	70	10
3	2	20				0	70	10

4)			$x_4=0$					
из т.1)			$x_2=1$			$x_2=0$		
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	4	40	1	40	4	0	0	0
4	3	20	0	40	4	0	0	0
5	6	30	1	70	10	1	30	6
1	5	30	0	70	10	1	30	11
3	2	20	0	70	10	1/2	70	12

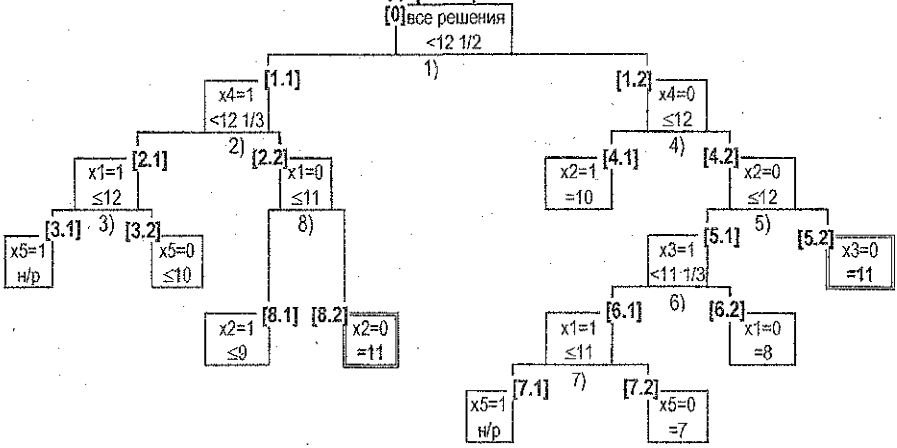
5)			$x_4=0,\ x_2=0$					
из т.4)			$x_3=1$			$x_3=0$		
i	c_i	t_i	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$	x_i	$\Sigma t_i\cdot x_i$	$\Sigma c_i\cdot x_i$
3	2	20	1	20	2	0	0	0
2	4	40	0	20	2	0	0	0
4	3	20	0	20	2	0	10	0
5	6	30	1	50	8	1	30	6
1	5	30	2/3	70	11 1/3	1	60	11☺

6)			$x_4=0,\ x_2=0,\ x_3=1$					
из т.5)			$x_1=1$			$x_1=0$		
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	5	30	1	30	5	0	0	0
3	2	20	1	50	7	1	20	2
2	4	40	0	50	7	0	20	2
4	3	20	0	50	7	0	20	2
5	6	30	2/3	70	11	1	50	8

7)			$x_4=0,\ x_2=0,\ x_3=1,\ x_1=1$					
из т.6)			$x_5=1$			$x_5=0$		
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	6	30	1	30		0	0	0
1	5	30	1	60		1	30	5
3	2	20	1	80		1	50	7
2	4	40	0	нет		0	50	7
4	3	20	0	решений		0	50	7

8)			$x_4=1,\ x_1=0$					
из т.2)			$x_2=1$			$x_2=0$		
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	4	40	1	40	4	0	0	0
1	5	30	0	40	4	0	0	0
4	3	20	1	60	7	1	20	3
5	6	30	1/3	70	9	1	30	9
3	2	20	0	70	9	1	70	11☺

Дерево решений:



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

Ответ: 1-е решение - $(x_1, x_2, x_3, x_4, x_5) = (1, 0, 0, 0, 1)$, общее время обработки равно 60;
2-е решение - $(x_1, x_2, x_3, x_4, x_5) = (0, 0, 1, 1, 1)$, общее время обработки равно 70.
Максимальная общая ценность выбранных для обработки деталей равна 11.

Примечание. Поясним построение дерева решений. При каждой вершине указан ее номер в квадратных скобках. Ветвления на дереве соответствуют таблицам.

На 1-м ветвлении ($x_4=1$, $x_4=0$) для дальнейшего рассмотрения выбираем вершину [1.1], где $x_4=1$, т.к. оценка суммарной ценности ($12\,^1/_3$) в этом случае больше, чем при $x_4=0$ (12). Из вершины [1.1] выполняем 2-е ветвление.

После 2-го ветвления открытыми остаются вершины [2.1], [2.2] и [1.2] с оценками суммарной ценности 12, 11 и 12 соответственно. Из них для дальнейшего ветвления выбираем любую из вершин с наибольшей оценкой (12), например, [2.1], где $x_1=1$.

После 3-го ветвления вершину [3.1] не рассматриваем: здесь при $x_4=x_1=x_5=1$ время обработки выбранных деталей превышает заданное – нет решений (н/р); открытыми остаются [3.2], [2.2] и [1.2]. Выбираем вершину [1.2], где есть шансы получить ценность, равную 12.

После 4-го ветвления в [4.1] при целочисленных x_i получаем ценность, равную 10. Для рассмотрения выбираем [4.2], где есть шансы получить бóльшую ценность, равную 12.

На 5-м ветвлении в вершине [5.2] получаем 1-е решение – это набор с ценностью, равной 11. Здесь получено точное значение ценности (при целочисленных x_i), которое является максимально возможным (см. открытые вершины [3.2], [2.2], [4.1], [5.1]).

Те вершины, где оценка ценности (10 в [3.2]) или ее точное значение (10 в [4.1]) меньше, чем уже найденное максимальное значение ценности (11 в [5.2]), далее не рассматриваем. В данный момент только в вершинах [2.2] и [5.1] остается шанс получить максимальную ценность, равную 11. Для дальнейшего ветвления выбираем [5.1] (с большей оценкой).

Аналогичным образом рассуждаем, выполняя оставшиеся ветвления на дереве решений.

1.3. Пример решения задачи о наилучшей загрузке станка в Excel

$$5 \cdot x_1 + 4 \cdot x_2 + 2 \cdot x_3 + 3 \cdot x_4 + 6 \cdot x_5 \to \max$$

$$30 \cdot x_1 + 40 \cdot x_2 + 20 \cdot x_3 + 20 \cdot x_4 + 30 \cdot x_5 \le 70,\ x_i \in \{0,1\},\ i=1,\ldots,5.$$

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)).

- Подготовим исходные данные на рабочем листе Excel.

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

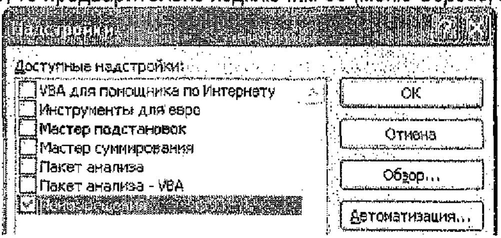
	A	B	C	D	E	F	G	H	I
1	i	1	2	3	4	5			
2	ci	5	4	2	3	6			
3	ti	30	40	20	20	30			
4	xi								

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

	A	B	C	D	E	F	G
1	i	1	2	3	4	5	
2	ci	5	4	2	3	6	=СУММПРОИЗВ(B2:F2;$B$4:$F$4)
3	ti	30	40	20	20	30	=СУММПРОИЗВ(B3:F3;$B$4:$F$4)
4	xi						

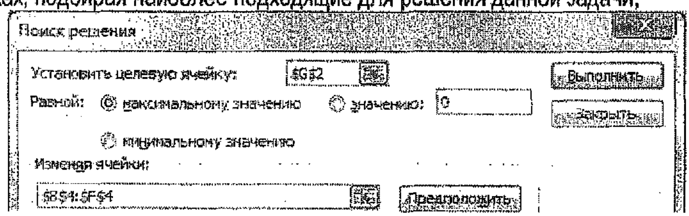
- Подготовим исходные данные в окне *«Поиск решения»*.

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



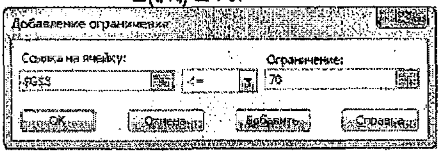
б) В окне *«Поиск решения»*:

- укажем *целевую ячейку* с формулой, задающей целевую функцию, чей максимум требуется найти (у нас это ячейка G2, 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,...,5 (у нас это диапазон ячеек B4:F4); так мы разрешаем Excel изменять значения в этих ячейках, подбирая наиболее подходящие для решения данной задачи;

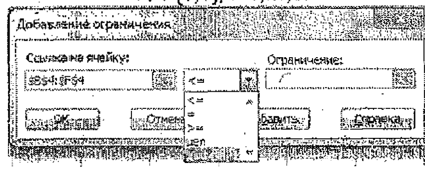


- добавим ограничения из математической модели задачи:

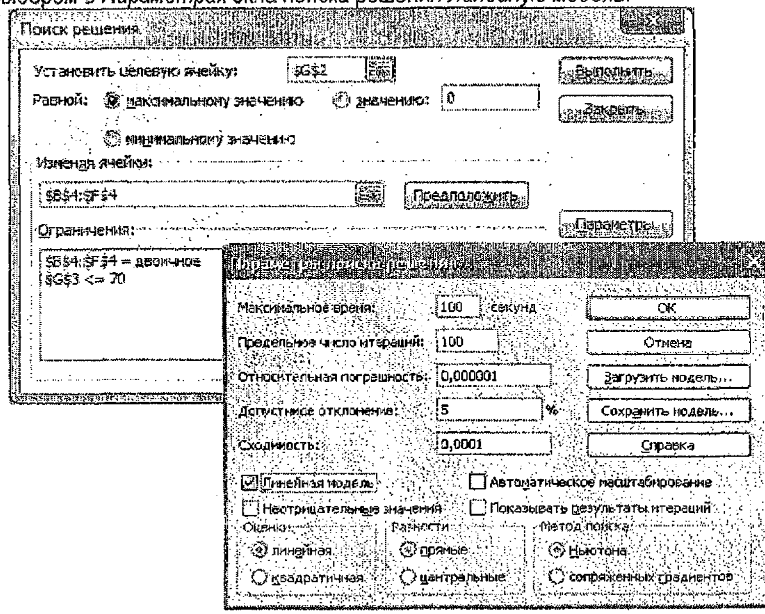
$\sum(t_i \cdot x_i) \le 70$:



$x_i \in \{0,1\}, i=1,\ldots,5$:

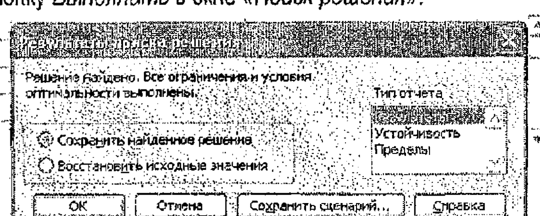


- выберем в *Параметрах* окна поиска решения *Линейную модель*:



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

	A	B	C	D	E	F	G
1	i	1	2	3	4	5	
2	ci	5	4	2	3	6	11
3	ti	30	40	20	20	30	60
4	xi	1	0	0	0	1	



Найдено одно из возможных решений задачи: в обработку берем 1-ю и 5-ю детали, т.к. $(x_1,x_2,x_3,x_4,x_5)=(1,0,0,0,1)$. Максимальная суммарная ценность выбранных деталей равна 11; при этом они могут быть обработаны за время, равное 60.

- Создадим *Отчет по результатам*, выбрав в появившемся окне «Результаты поиска решения» *Тип отчета - Результаты*. Созданный отчет появится на новом листе рабочей книги.

2.1. Составление расписания горячей обработки (РГО)

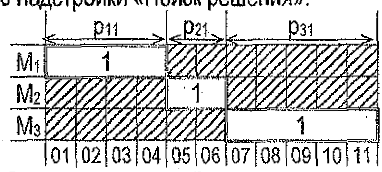
Задача. На линии горячей обработки, состоящей из k станков, нужно обработать партию из m деталей. Все детали должны проходить по линии в одном направлении через каждый станок. Заданы длительности обработки деталей каждым станком. Каждая деталь может ожидать обработку только перед первым станком линии, а перед остальными – ожидание недопустимо (войдя в линию, каждая деталь должна обрабатываться непрерывно – одним станком за другим). Требуется составить расписание горячей обработки данной партии деталей за минимальное время (минимальное по длине расписание).

Задания для выполнения.

1. По заданной матрице P получить матрицу W_0 размерности $(m+1)\times(m+1)$, где w_{ij} - время между моментами начала обработки 1-м станком деталей D_i и D_j по любому компактному расписанию, где детали идут одна за другой в порядке D_i, D_j. $(m+1)$-я деталь – фиктивная.
2. Для полученной матрицы W_0 найти оптимальный порядок обработки деталей за минимальное время. Построить диаграмму, соответствующую найденному порядку.
3. Решить поставленную задачу в Excel с помощью надстройки «Поиск решения».

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

$$P=\begin{bmatrix}4&6&1&3\\2&4&5&6\\5&2&4&2\end{bmatrix}$$



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 – 27 единиц:

Для 4-х деталей число всех компактных расписаний равно 4! (числу всех перестановок этих деталей). Минимальное по длине расписание совпадает с одним из компактных. Чтобы найти такое расписание, 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 $w_{12}=4$, $w_{23}=9$, $w_{34}=3$.

Чтобы учесть полное время обработки той из реальных деталей, которая обрабатывается последней, введем в рассмотрение фиктивную деталь, которая входит в линию последней и проходит обработку на всех станках мгновенно (за 0 ед. времени). Если в линии k станков, а число всех деталей равно m, то величины w_{ij} вычисляются 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 $m\times m$.

<u>Решение задачи.</u> Добавим в исходную матрицу *P* последний столбец, содержащий время обработки фиктивной детали каждым из станков (все элементы этого столбца равны 0). Получим матрицу величин w_{ij} по заданной матрице *P* в *Derive*. 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:=[4, 6, 1, 3, 0; 2, 4, 5, 6, 0; 5, 2, 4, 2, 0]

Нажав клавишу Enter, получим:

$$\#1:\ p := \begin{bmatrix} 4 & 6 & 1 & 3 & 0 \\ 2 & 4 & 5 & 6 & 0 \\ 5 & 2 & 4 & 2 & 0 \end{bmatrix}$$

Далее вводим выражение для получения матрицы W:

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, 3)), j, 1, 5), i, 1, 5),

где 3 – число станков, а 5 – число деталей, включая фиктивную. Нажав Ctrl+↵, получим:

$$\#2:\ \mathrm{VECTOR}\left(\mathrm{VECTOR}\left(\mathrm{MAX}\left(\mathrm{VECTOR}\left(\left(\sum_{s=1}^{h} p_{s,i}\right) - \sum_{s=1}^{h-1} p_{s,j}, h, 1, 3\right)\right), j, 1, 5\right), i, 1, 5\right)$$

$$\#3\quad \begin{bmatrix} 5 & 4 & 5 & 4 & 11 \\ 6 & 6 & 9 & 7 & 12 \\ 4 & 1 & 5 & 3 & 10 \\ 5 & 3 & 8 & 6 & 11 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

Запишем полученную матрицу, заменив элементы главной диагонали знаком ∞ :

$$W_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 4 & 5 & 4 & 11 \\ 6 & \infty & 9 & 7 & 12 \\ 4 & 1 & \infty & 3 & 10 \\ 5 & 3 & 8 & \infty & 11 \\ 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array}$$

Значения 5-го столбца можно вычислить по матрице P:
4+2+5=11 – время обработки 1-й детали на всех станках;
6+4+2=12 – время обработки 2-й детали на всех станках;
1+5+4=10 – время обработки 3-й детали на всех станках;
5+3+8=11 – время обработки 4-й детали на всех станках.
Значения 5-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 <u>методом ветвей и границ</u>. Приведем <u>алгоритм</u>, 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).

<u>Шаг 1</u> в нашем примере:

$$W_0 = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 4 & 5 & 4 & 11 \\ 6 & \infty & 9 & 7 & 12 \\ 4 & 1 & \infty & 3 & 10 \\ 5 & 3 & 8 & \infty & 11 \\ 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array}$$

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

Если минимальным элементом строки (столбца) является 0, то его вычитание не изменит строку (столбец). Поэтому выписываем только ненулевые минимальные элементы. Вычитая из каждой строки минимальный элемент этой строки, из W_0 получим преобразованную матрицу (слева):

$$\begin{array}{c|c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} \infty & 0 & 1 & 0 & 7 \\ 0 & \infty & 3 & 1 & 6 \\ 3 & 0 & \infty & 2 & 9 \\ 2 & 0 & 5 & \infty & 8 \\ 0 & 0 & 0 & 0 & \infty \end{bmatrix} \\ & \begin{array}{ccccc} & & & & 6 \end{array} \end{array}$$

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).

$$W_0^* = \begin{array}{c|c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} \infty & 0 & 1 & 0 & 1 \\ 0 & \infty & 3 & 1 & 0 \\ 3 & 0 & \infty & 2 & 3 \\ 2 & 0 & 5 & \infty & 2 \\ 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array}$$

2. Сумма всех вычтенных чисел даст нижнюю оценку C_0 общего времени обработки деталей. В ходе решения величина C_0 может только увеличиться или остаться прежней.

Шаг 2 в нашем примере: $C_0 = \underbrace{4+6+1+3}_{\text{то, что вычли по строкам}} + \underbrace{6}_{\text{то, что вычли по столбцам}} = 20.$

3. Если размерность рассматриваемой приведенной матрицы больше 2-х, то в этой матрице 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!

Если размерность рассматриваемой приведенной матрицы W_{t1}^* 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_{t1}^*. 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_{t1}. Последовательность начнется с той детали, которая следует непосредственно за фиктивной, и закончится той деталью, которая предшествовала фиктивной 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_{t1} - минимальному времени, за которое можно обработать всю партию деталей на линии горячей обработки.

Шаг 3 в примере. Размерность W_0^* больше 2-х, значит, вычисляем степени нулей.

$$W_0^* = \begin{array}{c|c} & \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{array}{c} 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} & \begin{bmatrix} \infty & 0^0 & 1 & 0^0 & 1 \\ 0^0 & \infty & 3 & 1 & 0^1 \\ 3 & 0^2 & \infty & 2 & 3 \\ 2 & 0^2 & 5 & \infty & 2 \\ 0^0 & 0^0 & 0^1 & 0^0 & \infty \end{bmatrix} \end{array}$$

Поясним на примере, каким образом вычисляются степени нулей. Степень нуля, стоящего в 4-й строке и 2-м столбце, равна 2+0=2, т.к. минимальным элементом 4-й строки (кроме рассматриваемого нуля) является 2, а минимальным элементом 2-го столбца (кроме рассматриваемого нуля) является 0 (любой из оставшихся в этом 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 4-й строки и 2-го столбца. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход $4 \to 2$, а второе запрещает этот переход: $4 \not\to 2$.

5. Для случая $i \to j$ строим из рассматриваемой приведенной матрицы новую матрицу $W_{t+1,1}$ следующим образом: в рассматриваемой приведенной матрице удаляем строку i и столбец j. При этом в новой матрице $W_{t+1,1}$ запрещаем переход, образующий цикл, в котором не все детали участвуют (для этого соответствующий элемент матрицы заменяем знаком ∞). Ситуации, которые возникают при добавлении нового перехода:

- 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 $\boxed{i' \ldots \to i}$ и $\boxed{i \to j}$ получаем $\boxed{i' \ldots \to i \to j}$ (соединяем по i);
либо так: из $\boxed{j \to \ldots i'}$ и $\boxed{i \to j}$ получаем $\boxed{i \to j \to \ldots i'}$ (соединяем по j);
либо так: из $\boxed{j \to \ldots i'}$, $\boxed{j' \ldots \to i}$ и $\boxed{i \to j}$ получим $\boxed{j' \ldots \to i \to j \to \ldots i'}$ (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 $\boxed{i' \ldots \to i \to j}$ заменяем знаком ∞ элемент в строке j и столбце i';
для цепочки вида $\boxed{i \to j \to \ldots i'}$ заменяем знаком ∞ элемент в строке i' и столбце i;
для цепочки вида $\boxed{j' \ldots \to i \to j \to \ldots i'}$ заменяем знаком ∞ эл-т в строке i' и столбце j'.

6. Для случая $i \not\to j$ строим из рассматриваемой приведенной матрицы новую матрицу $W_{t+1,2}$: переписываем рассматриваемую приведенную матрицу и запрещаем в ней переход из i в j, т.е. элемент в строке i и столбце j заменяем знаком ∞.

$4 \to 2$

$$W_{11} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 1 & 0 & 1 \\ 0 & 3 & \infty & 0 \\ 3 & \infty & 2 & 3 \\ 0 & 0 & 0 & \infty \end{bmatrix} \end{array} 2$$

В примере выполняем шаги 5 и 6 алгоритма:
Шаг 5. В матрице W_{11}, полученной из W_0^* вычеркиванием 4-й строки и 2-го столбца, запрещаем знаком ∞ переход из 2 в 4 (по концам дуги $4 \to 2$ в противоположном направлении, чтобы далее не образовался цикл, не содержащий обозначения всех пяти деталей).

$4 \not\to 2$

$$W_{12} = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array} \begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 0 & 1 & 0 & 1 \\ 0 & \infty & 3 & 1 & 0 \\ 3 & 0 & \infty & 2 & 3 \\ 2 & \infty & 5 & \infty & 2 \\ 0 & 0 & 0 & 0 & \infty \end{bmatrix} \end{array} 2$$

Шаг 6. В матрице W_{12}, полученной переписыванием W_0^*, запрещаем знаком ∞ переход из 4 в 2 (т.к. рассматривается ситуация, когда переход из 4 в 2 запрещен).

7. По матрице $W_{t+1,1}$ получаем оценку $C_{t+1,1}$, которая показывает нижнюю оценку общего времени обработки деталей в случае, если после детали i обрабатывается деталь j.

Для этого находим сумму, которую нужно вычесть из строк и столбцов матрицы $W_{t+1,1}$, чтобы получить соответствующую приведенную матрицу $W_{t+1,1}^*$ (см. шаг 1 алгоритма). Эта сумма, добавленная к оценке, полученной на предыдущем шаге, дает $C_{t+1,1}$.

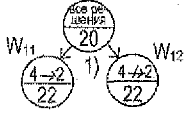
8. По матрице $W_{t+1,2}$ получаем оценку $C_{t+1,2}$, которая показывает нижнюю оценку общего времени обработки деталей, если после детали *i* не обрабатывается деталь *j*. Для этого находим сумму, которую нужно вычесть из строк и столбцов матрицы $W_{t+1,2}$, чтобы получить соответствующую приведенную матрицу $W_{t+1,2}^*$ (см. шаг 1 алгоритма). Эта сумма, добавленная к оценке, полученной на предыдущем шаге, дает $C_{t+1,2}$.
9. Рассматриваем оценки $C_{t+1,1}$, $C_{t+1,2}$, 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) и рассматривать соответствующую ей приведенную. Если имеется несколько одинаковых наименьших оценок, то из них выбираем ту, которая соответствует матрице, имеющей меньшую размерность. Эту матрицу и выбираем для дальнейшего рассмотрения. Полагаем $t := t+1$ (увеличиваем t на 1) и переходим к шагу 3 алгоритма.

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

Шаг 7. Матрица W_{11} содержит нули в каждом из столбцов, но в 4-й строке нет нулевых элементов, поэтому, чтобы получить приведенную матрицу W_{11}^*, необходимо вычесть из 4-й строки ее минимальный элемент, равный 2. Вычтенное число (2) добавим к предыдущей оценке C_0, равной 20. Получим оценку $C_{11}=20+2=22$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если сразу после детали номер 4 будет обрабатываться деталь номер 2 ($4 \to 2$).

Шаг 8. Матрица W_{12} содержит нули в каждом из столбцов, но в 4-й строке нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{12}^*, необходимо вычесть из 4-й строки ее минимальный элемент, равный 2. Вычтенное число (2) добавим к предыдущей оценке C_0, равной 20. Получим оценку $C_{12}=20+2=22$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку 4-й детали непосредственно после 2-й детали ($4 \not\to 2$).

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.



Шаг 9. Из двух открытых вершин выбираем ту, которой соответствует меньшая оценка (т.к. нас интересует минимальное время обработки всех деталей). В нашем случае обе наименьшие оценки одинаковы ($C_{11}=C_{12}=22$). В таком случае выбираем вершину, которой соответствует матрица меньшей размерности – это матрица W_{11}. $t := 0+1=1$. Возвращаемся к шагу 3 алгоритма и рассматриваем приведенную матрицу W_{11}^*.

$$W_{11}^* = \begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \end{array} \begin{array}{c} \begin{array}{cccc} 1 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 1 & 0^1 & 1 \\ 0^0 & 3 & \infty & 0^1 \\ 1 & \infty & 0^1 & 1 \\ 0^0 & 0^1 & 0^0 & \infty \end{bmatrix} \end{array}$$

Шаг 3. Размерность матрицы W_{11}^* больше двух, поэтому вычисляем степени нулей этой матрицы.

Шаг 4. Выбираем ноль с наибольшей степенью. Таких нулевых элементов с максимальной степенью, равной 1, несколько. Можем из них выбрать любой, например, стоящий в 3-й строке и 4-м столбце.

Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход $3\to4$ (при условии, что переход из 4 в 2 разрешен), а второе запрещает переход: $3\not\to4$ (при таком же условии: $4\to2$).
Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_{11}^* найдем матрицы W_{21} (для $3\to4$) и W_{22} (для $3\not\to4$).

$3\to4\to2$

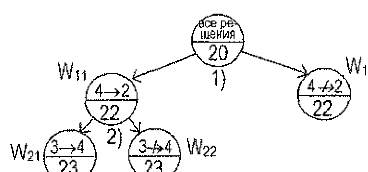
$$W_{21}=\begin{array}{c} \\ 1 \\ 2 \\ 5 \end{array}\!\!\begin{array}{c} \begin{array}{ccc} 1 & 3 & 5 \end{array} \\ \begin{bmatrix} \infty & 1 & 1 \\ 0 & \infty & 0 \\ 0 & 0 & \infty \end{bmatrix} \end{array}\!\!\begin{array}{c} \\ 1 \\ \\ \\ \end{array}$$

Шаг 5. В матрице W_{21}, которая получена из W_{11}^* вычеркиванием 3-й строки и 4-го столбца, запрещаем переход из 2 в 3: т.к. добавляемый переход $3\to4$ имеет общую вершину 4 с переходом $4\to2$, добавленным ранее, то, соединив их по этой общей вершине, получим цепочку $3\to4\to2$, по концам которой из 2 в 3 запрещаем переход, чтобы не получить цикл, содержащий не все детали.

$4\to2$, $3\not\to4$

$$W_{22}=\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 5 \end{array}\!\!\begin{array}{c} \begin{array}{cccc} 1 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 1 & 0 & 1 \\ 0 & 3 & \infty & 0 \\ 1 & \infty & \infty & 1 \\ 0 & 0 & 0 & \infty \end{bmatrix} \end{array}\!\!\begin{array}{c} \\ \\ \\ 1 \\ \\ \end{array}$$

Шаг 6. В матрице W_{22}, полученной переписыванием W_{11}^*, запрещаем переход из 3 в 4.
Выполним для нашего примера шаги 7, 8 и 9 алгоритма: уточним оценки для W_{21} и W_{22} и выберем наименьшую из оценок, соответствующих висячим вершинам на дереве.
Шаг 7. Матрица W_{21} не содержит 0 в 1-й строке, поэтому, чтобы получить приведенную матрицу W_{21}^*, вычитаем из 1-й строки ее минимальный элемент (1). Это число добавим к предыдущей оценке, равной 22. Получим оценку $C_{21}=22+1=23$ – нижнюю оценку общего времени обработки всех деталей, если сразу после D_3 обрабатывать D_4 ($3\to4$), при этом $4\to2$.
Шаг 8. Матрица W_{22} не содержит нуля в 3-й строке, поэтому, чтобы получить приведенную матрицу W_{22}^*, надо вычесть из 3-й строки ее минимальный элемент, равный 1. Вычтенное число (1) добавим к предыдущей оценке, равной 22. Получим оценку $C_{22}=22+1=23$. Она дает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали D_4 сразу после детали D_3 ($3\not\to4$), при условии, что $4\to2$.
Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{11}), выбранной на предыдущем этапе.



Шаг 9. Из оценок $C_{21}=23$, $C_{22}=23$, $C_{12}=22$, соответствующих висячим вершинам дерева, выбираем наименьшую. Наименьшей является оценка $C_{12}=22$. Далее будем работать с матрицей, для которой получена эта оценка, т.е. с матрицей W_{12}. $t := 1+1=2$.
Переходим к шагу 3 алгоритма и рассматриваем приведенную матрицу W_{12}^*.

$$W_{12}^*=\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{array}\!\!\begin{array}{c} \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 0^0 & 1 & 0^0 & 1 \\ 0^0 & \infty & 3 & 1 & 0^0 \\ 3 & 0^2 & \infty & 2 & 3 \\ 0^0 & \infty & 3 & \infty & 0^0 \\ 0^0 & 0^0 & 0^1 & 0^0 & \infty \end{bmatrix} \end{array}$$

Шаг 3. Размерность W_{12}^* больше 2-х, поэтому ищем степени нулей.
Шаг 4. Единственный нулевой элемент с максимальной степенью, равной 2, находится в 3-й строке и 2-м столбце. Значит, множество всех решений разобьется на два непересекающихся подмножества: первое содержит переход $3\to2$ (при условии, что $4\not\to2$), а второе запрещает этот переход: $3\not\to2$ (при том же условии).

$$\begin{array}{c} 4\not\to 2,\ 3\to 2 \\ W_{31}= \begin{array}{c|cccc} & 1 & 3 & 4 & 5 \\ \hline 1 & \infty & 1 & 0 & 1 \\ 2 & 0 & \infty & 1 & 0 \\ 4 & 0 & 3 & \infty & 0 \\ 5 & 0 & 0 & 0 & \infty \end{array} \end{array}$$

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_{12}^* найдем матрицы W_{31} (для $3\to 2$) и W_{32} (для $3\not\to 2$).

Шаг 5. В матрице W_{31}, которая получена из W_{12}^* вычеркиванием 3-й строки и 2-го столбца, знаком ∞ запрещаем переход из 2 в 3.

Шаг 6. В матрице W_{32}, полученной переписыванием W_{12}^*, запрещаем переход из 3 в 2.

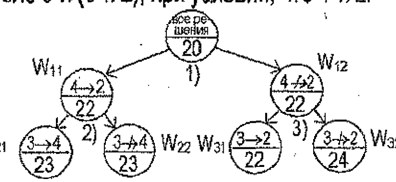
$$\begin{array}{c} 4\not\to 2,\ 3\not\to 2 \\ W_{32}= \begin{array}{c|ccccc|c} & 1 & 2 & 3 & 4 & 5 & \\ \hline 1 & \infty & 0 & 1 & 0 & 1 & \\ 2 & 0 & \infty & 3 & 1 & 0 & \\ 3 & 3 & \infty & \infty & 2 & 3 & 2 \\ 4 & 0 & \infty & 3 & \infty & 0 & \\ 5 & 0 & 0 & 0 & 0 & \infty & \end{array} \end{array}$$

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, равной 22, и дает нижнюю оценку общего времени обработки всех деталей в случае, если сразу же после детали с номером 3 будет обрабатываться деталь с номером 2 ($3\to 2$), при условии, что $4\not\to 2$.

Шаг 8. Матрица W_{32} содержит нули во всех строках, кроме 3-й, значит, чтобы получить приведенную матрицу W_{32}^*, нужно вычесть из 3-й строки ее минимальный элемент, равный 2. Вычтенное число добавим к предыдущей оценке, равной 22. Получим $C_{32}=22+2=24$, дающую нижнюю оценку общего времени обработки всех деталей, если запретить обработку 2-й детали сразу после 3-й ($3\not\to 2$), при условии, что $4\not\to 2$.

Наращиваем дерево решений из висячей вершины с меньшей оценкой (C_{12}), выбранной на предыдущем этапе.

Шаг 9. Из оценок $C_{21}=23$, $C_{22}=23$, $C_{31}=22$, $C_{32}=24$, соответствующих висячим вершинам дерева, выбираем наименьшую.



Это оценка $C_{31}=22$. Далее работаем с соответствующей матрицей W_{31}. $t:=2+1=3$. Переходим к шагу 3 алгоритма и рассматриваем приведенную матрицу W_{31}^*.

$$W_{31}^*= \begin{array}{c|cccc} & 1 & 3 & 4 & 5 \\ \hline 1 & \infty & 1 & 0^1 & 1 \\ 2 & 0^0 & \infty & 1 & 0^0 \\ 4 & 0^0 & 3 & \infty & 0^0 \\ 5 & 0^0 & 0^1 & 0^0 & \infty \end{array}$$

Шаг 3. Размерность W_{31}^* больше двух, поэтому ищем степени нулей.

Шаг 4. Нулей с максимальной степенью, равной 1, два. Выберем, например, ноль в 5-й строке и 3-м столбце. Тогда мн-во всех решений разобьется на два непересекающихся подмножества: 1-е содержит переход $5\to 3$ (при этом $4\not\to 2$, $3\to 2$), 2-е запрещает этот переход: $5\not\to 3$.

Выполним далее для нашего примера шаги 5 и 6 алгоритма, т.е. по матрице W_{31}^* найдем матрицы W_{41} (для $5\to 3$) и W_{42} (для $5\not\to 3$).

$$\begin{array}{c} 4\not\to 2,\ 5\to 3\to 2 \\ W_{41}= \begin{array}{c|ccc} & 1 & 4 & 5 \\ \hline 1 & \infty & 0 & 1 \\ 2 & 0 & 1 & \infty \\ 4 & 0 & \infty & 0 \end{array} \end{array}$$

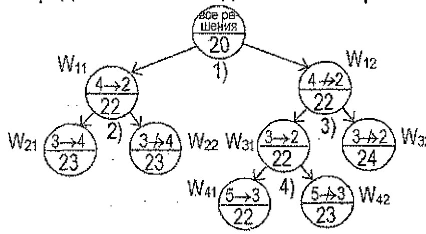
Шаг 5. В матрице W_{41}, полученной из W_{31}^* вычеркиванием 5-й строки и 3-го столбца, запрещаем переход из 2 в 5. Т.к. добавляемый переход $5\to 3$ имеет общую вершину 3 с цепочкой $3\to 2$, полученной ранее, то, соединив их по вершине 3, получим цепочку $5\to 3\to 2$; по концам ее из 2 в 5 запрещаем переход.

$$\begin{array}{c} 4\not\to 2,\ 3\to 2,\ 5\not\to 3 \\ W_{42}= \begin{array}{c|cccc} & 1 & 3 & 4 & 5 \\ \hline 1 & \infty & 1 & 0 & 1 \\ 2 & 0 & \infty & 1 & 0 \\ 4 & 0 & 3 & \infty & 0 \\ 5 & 0 & \infty & 0 & \infty \end{array} \end{array}$$

Шаг 6. В матрице W_{42}, полученной переписыванием W_{31}^*, запрещаем переход из 5 в 3.

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, равной 22, и дает нижнюю оценку общего времени обработки всех деталей, если сразу же после детали 5 будет обрабатываться деталь 3 ($5 \to 3$), учитывая, что $3 \to 2$, $4 \not\to 2$.

Шаг 8. Матрица W_{42} содержит нули в каждом из столбцов, но в столбце с номером 3 нет нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{42}^*, необходимо вычесть из 3-го столбца его минимальный элемент, равный 1. Вычтенное число (1) добавим к предыдущей оценке, равной 22. Получим оценку $C_{42}=22+1=23$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 3 непосредственно после детали с номером 5 ($5 \not\to 3$), при условии, что $4 \not\to 2$ и $3 \to 2$.



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

Шаг 9. Из оценок $C_{21}=C_{22}=C_{42}=23$, $C_{32}=24$, $C_{41}=22$ выбираем наименьшую. Это оценка $C_{41}=22$. Далее будем работать с соответствующей матрицей, т.е. с W_{41}. $t := 3+1=4$.

Переходим к шагу 3 алгоритма. Рассмотрим приведенную матрицу W_{41}^*.

Шаг 3. Размерность W_{41}^* больше двух, поэтому ищем степени нулей.

$$W_{41}^* = \begin{array}{c} \\ 1 \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{ccc} 1 & 4 & 5 \end{array} \\ \begin{bmatrix} \infty & 0^2 & 1 \\ 0^1 & 1 & \infty \\ 0^0 & \infty & 0^1 \end{bmatrix} \end{array}$$

Шаг 4. Выбираем 0 с максимальной степенью, равной 2, который находится во 1-й строке и 4-м столбце. Мн-во всех решений разобьется на два подмножества: 1-е содержит переход $1 \to 4$ (при условии, что $5 \to 3 \to 2$ и $4 \not\to 2$), а 2-е запрещает этот переход: $1 \not\to 4$ (при тех же условиях).

Выполним шаги 5 и 6, т.е. по W_{41}^* найдем матрицы W_{51} (для $1 \to 4$) и W_{52} (для $1 \not\to 4$).

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

$$W_{51} = \begin{array}{c} \\ 2 \\ 4 \end{array} \begin{array}{c} \begin{array}{cc} 1 & 5 \end{array} \\ \begin{bmatrix} 0 & \infty \\ \infty & 0 \end{bmatrix} \end{array}$$

Шаг 5. В матрице W_{51}, полученной из W_{41}^* вычеркиванием 1-й строки и 4-го столбца, запрещаем переход из 4 в 1: т.к. добавляемый переход $1 \to 4$ не имеет общих вершин с полученной ранее цепочкой $5 \to 3 \to 2$, то запрещаем переход по концам вновь добавляемой дуги $1 \to 4$ в противоположном направлении, из 4 в 1.

$4 \not\to 2$, $1 \not\to 4$

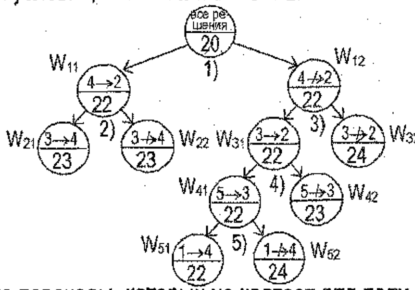
$$W_{52} = \begin{array}{c} \\ 1 \\ 2 \\ 4 \\ \\ \end{array} \begin{array}{c} \begin{array}{ccc} 2 & 4 & 6 \end{array} \\ \begin{bmatrix} \infty & \infty & 1 \\ 0 & 1 & \infty \\ 0 & \infty & 0 \end{bmatrix} \\ \begin{array}{ccc} & 1 & \end{array} \end{array} \begin{array}{c} \\ 1 \\ \\ \\ \\ \end{array}$$

Шаг 6. В матрице W_{52}, полученной переписыванием W_5^*, запрещаем переход из 1 в 4.

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, равной 22, и показывает общее время обработки всех деталей, если сразу же после детали номер 1 будет обрабатываться деталь номер 4 ($1 \to 4$), при условии, что $4 \not\to 2$ и $5 \to 3 \to 2$.

Шаг 8. Матрица W_{52} в 1-й строке не содержит нулевого элемента, поэтому, чтобы получить приведенную матрицу W_{52}^*, необходимо вычесть из 1-й строки ее минимальный элемент, равный 1. После этого действия в W_{52} остается 4-й столбец, не содержащий нуля. Вычитаем из 4-го столбца его минимальный элемент (1). Получим оценку $C_{52}=22+1+1=24$, которая показывает нижнюю оценку общего времени обработки всех деталей в случае, если запретить обработку детали с номером 4 непосредственно после детали с номером 1 ($1 \not\to 4$), при условии, что $4 \not\to 2$ и $5 \to 3 \to 2$.

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



Шаг 3. Размерность W_{51}^* равна двум. На этом работа алгоритма завершена.

$$1\to4,\ 5\to3\to2$$

$$W_{51}^* = \begin{array}{c|cc} & 1 & 5 \\ \hline 2 & 0 & \infty \\ 4 & \infty & 0 \end{array}$$

По матрице W_{51}^* видно, что в ней разрешены те переходы, которых не хватает для получения из цепочек $1\to4$ и $5\to3\to2$ полного цикла: $2\to1$ и $4\to5$. Добавим переходы $4\to5$ и $2\to1$ к построенным ранее цепочкам $1\to4$ и $5\to3\to2$ и получим цикл, в котором участвуют все детали: $\underbrace{1\to4\to5\to3\to2}$. При этом полное минимальное время обработки этих деталей в указанном порядке составляет $C_{51}=22$ единицы. Проверим это, используя исходную матрицу W_0. Каждому переходу $i\to j$ в найденном цикле соответствует число – элемент w_{ij} матрицы W_0.

$$\underbrace{1\xrightarrow{4}4\xrightarrow{11}5\xrightarrow{0}3\xrightarrow{1}2}_{6}$$

Найдем их сумму: 4+11+0+1+6=22 – общее время обработки получено правильно.

Но в полученном цикле присутствует фиктивная 5-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 четырех деталей за минимальное время, равное 22: $\underbrace{1\to4\cancel{\to5\to}3\to2}$.

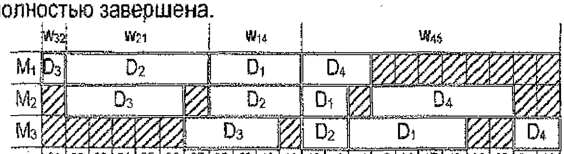
$$D_3\to D_2\to D_1\to D_4$$ – найденный порядок обработки.

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

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

$$P = \begin{array}{c|cccc} & D_1 & D_2 & D_3 & D_4 \\ \hline M_1 & 4 & 6 & 1 & 3 \\ M_2 & 2 & 4 & 5 & 6 \\ M_3 & 5 & 2 & 4 & 2 \end{array} \qquad W_0 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & \infty & 4 & 5 & 4 & 11 \\ 2 & 6 & \infty & 9 & 7 & 12 \\ 3 & 4 & 1 & \infty & 3 & 10 \\ 4 & 5 & 3 & 8 & \infty & 11 \\ 5 & 0 & 0 & 0 & 0 & \infty \end{array}$$

По матрице W_0 определяем время, которое проходит от начала обработки одной детали до начала обработки следующей за ней детали первым станком. Имеем: $w_{32}=1$, $w_{21}=6$, $w_{14}=4$, $w_{45}=11$. При построении 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 22 единицы.

2.2. Пример оформления решения задачи о составлении РГО

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

<u>Решение.</u> Матрица P, задающая длительности обработки деталей каждым станком:

	D_1	D_2	D_3	D_4
M_1	4	6	1	3
M_2	2	4	5	6
M_3	5	2	4	2

Запишем матрицу величин w_{ij}, полученную в *Derive* по матрице P с добавленным 5-м столбцом для фиктивной детали, а элементы главной диагонали заменим знаком ∞:

$$W_0=\begin{array}{c} \\ 1\\2\\3\\4\\5\\ \\ \end{array}\begin{array}{c} \begin{array}{ccccc}1&2&3&4&5\end{array}\\ \begin{bmatrix}\infty&4&5&4&11\\6&\infty&9&7&12\\4&1&\infty&3&10\\5&3&8&\infty&11\\0&0&0&0&\infty\end{bmatrix}\\ \begin{array}{ccccc}&&&&6\end{array}\end{array}\begin{array}{c} \\ 4\\6\\1\\3\\ \\ \\ \end{array}$$

Приведенная матрица W_0^* (справа):

$$W_0^*=\begin{array}{c} \\ 1\\2\\3\\4\\5\end{array}\begin{array}{c} \begin{array}{ccccc}1&2&3&4&5\end{array}\\ \begin{bmatrix}\infty&0&1&0&1\\0&\infty&3&1&0\\3&0&\infty&2&3\\2&0&5&\infty&2\\0&0&0&0&\infty\end{bmatrix}\end{array}$$

$C_0=4+6+1+3+6=20$

Строим корневую вершину дерева:

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

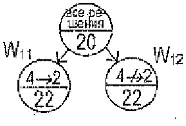
$$W_0^*=\begin{array}{c} \\ 1\\2\\3\\4\\5\end{array}\begin{array}{c} \begin{array}{ccccc}1&2&3&4&5\end{array}\\ \begin{bmatrix}\infty&0^0&1&0^0&1\\0^0&\infty&3&1&0^1\\3&0^2&\infty&2&3\\2&0^2&5&\infty&2\\0^0&0^0&0^1&0^0&\infty\end{bmatrix}\end{array}$$

Рассматриваем 2 случая – **$4\to2$** и **$4\not\to2$** и строим очередной ярус дерева решений:

$4\to2$

$$W_{11}=\begin{array}{c} \\ 1\\2\\3\\5\end{array}\begin{array}{c} \begin{array}{cccc}1&3&4&5\end{array}\\ \begin{bmatrix}\infty&1&0&1\\0&3&\infty&0\\3&\infty&2&3\\0&0&0&\infty\end{bmatrix}\end{array}\begin{array}{c} \\ \\ \\ 2\\ \\ \end{array}$$

$C_{11}=20+2=22$



$4\not\to2$

$$W_{12}=\begin{array}{c} \\ 1\\2\\3\\4\\5\end{array}\begin{array}{c} \begin{array}{ccccc}1&2&3&4&5\end{array}\\ \begin{bmatrix}\infty&0&1&0&1\\0&\infty&3&1&0\\3&0&\infty&2&3\\2&\infty&5&\infty&2\\0&0&0&0&\infty\end{bmatrix}\end{array}\begin{array}{c} \\ \\ \\ \\ 2\\ \\ \end{array}$$

$C_{12}=20+2=22$

Т.к. оценки обеих висячих вершин одинаковы, а размерность матрицы W_{11} меньше, чем размерность матрицы W_{12}, то переходим к приведенной матрице W_{11}^*:

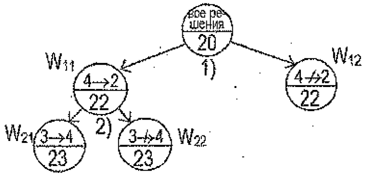
$$W_{11}^*=\begin{array}{c} \\ 1\\2\\3\\5\end{array}\begin{array}{c} \begin{array}{cccc}1&3&4&5\end{array}\\ \begin{bmatrix}\infty&1&0^1&1\\0^0&3&\infty&0^1\\1&\infty&0^1&1\\0^0&0^1&0^0&\infty\end{bmatrix}\end{array}$$

Рассматриваем 2 случая – **3→4** и **3↛4** и строим очередной ярус дерева решений:

$$3\to4\to2$$

$$W_{21}=\begin{array}{cc} & \begin{matrix}1&3&5\end{matrix}\\ \begin{matrix}1\\2\\5\end{matrix} & \begin{bmatrix}\infty&1&1\\0&\infty&0\\0&0&\infty\end{bmatrix}\begin{matrix}1\\ \\ \\ \end{matrix}\end{array}$$

$C_{21}=22+1=23$



$$4\to2,\ 3\not\to4$$

$$W_{22}=\begin{array}{cc} & \begin{matrix}1&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\3\\5\end{matrix} & \begin{bmatrix}\infty&1&0&1\\0&3&\infty&0\\1&\infty&\infty&1\\0&0&0&\infty\end{bmatrix}\begin{matrix} \\ \\1\\ \end{matrix}\end{array}$$

$C_{22}=22+1=23$

Т.к. оценка для матрицы W_{12} наименьшая, переходим к приведенной матрице W_{12}^{*}:

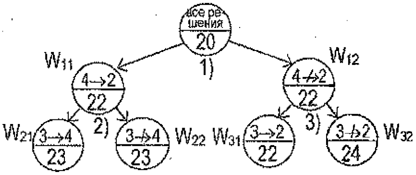
$$W_{12}^{*}=\begin{array}{cc} & \begin{matrix}1&2&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}\infty&0^0&1&0^0&1\\0^0&\infty&3&1&0^0\\3&0^2&\infty&2&3\\0^0&\infty&3&\infty&0^0\\0^0&0^0&0^1&0^0&\infty\end{bmatrix}\end{array}$$

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

$$4\not\to2,\ 3\to2$$

$$W_{31}=\begin{array}{cc} & \begin{matrix}1&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\4\\5\end{matrix} & \begin{bmatrix}\infty&1&0&1\\0&\infty&1&0\\0&3&\infty&0\\0&0&0&\infty\end{bmatrix}\end{array}$$

$C_{31}=22+0=22$



$$4\not\to2,\ 3\not\to2$$

$$W_{32}=\begin{array}{cc} & \begin{matrix}1&2&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\3\\4\\5\end{matrix} & \begin{bmatrix}\infty&0&1&0&1\\0&\infty&3&1&0\\3&\infty&\infty&2&3\\0&\infty&3&\infty&0\\0&0&0&0&\infty\end{bmatrix}\begin{matrix} \\ \\2\\ \\ \end{matrix}\end{array}$$

$C_{32}=22+2=24$

Т.к. оценка для матрицы W_{31} наименьшая, то переходим к приведенной матрице W_{31}^{*}:

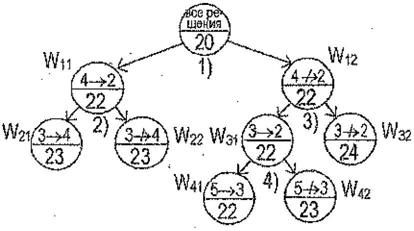
$$W_{31}^{*}=\begin{array}{cc} & \begin{matrix}1&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\4\\5\end{matrix} & \begin{bmatrix}\infty&1&0^1&1\\0^0&\infty&1&0^0\\0^0&3&\infty&0^0\\0^0&0&0^0&\infty\end{bmatrix}\end{array}$$

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

$$4\not\to2,\ 5\to3\to2$$

$$W_{41}=\begin{array}{cc} & \begin{matrix}1&4&5\end{matrix}\\ \begin{matrix}1\\2\\4\end{matrix} & \begin{bmatrix}\infty&0&1\\0&1&\infty\\0&\infty&0\end{bmatrix}\end{array}$$

$C_{41}=22+0=22$



$$4\not\to2,\ 3\to2,\ 5\not\to3$$

$$W_{42}=\begin{array}{cc} & \begin{matrix}1&3&4&5\end{matrix}\\ \begin{matrix}1\\2\\4\\5\end{matrix} & \begin{bmatrix}\infty&1&0&1\\0&\infty&1&0\\0&3&\infty&0\\0&\infty&0&\infty\end{bmatrix}\\ & \begin{matrix}&1&&\end{matrix}\end{array}$$

$C_{42}=22+1=23$

Оценка для W_{41} наименьшая, значит, переходим к W_{41}^*:

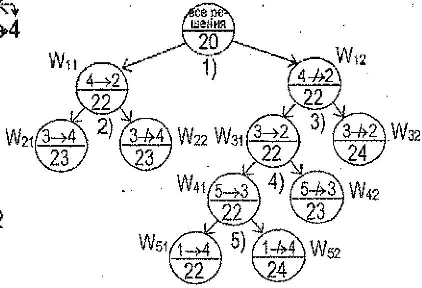
$$W_{41}^* = \begin{array}{c|ccc} & 1 & 4 & 5 \\ \hline 1 & \infty & 0^2 & 1 \\ 2 & 0^1 & 1 & \infty \\ 4 & 0^0 & \infty & 0^1 \end{array}$$

Рассматриваем 2 случая – **1→4** и **1↛4**, и строим очередной ярус дерева решений:

4↛2, 5→3→2, 1→4

$$W_{51} = \begin{array}{c|cc} & 1 & 5 \\ \hline 2 & 0 & \infty \\ 4 & \infty & 0 \end{array}$$

$C_{51}=22+0=22$



4↛2, 5→3→2, 1↛4

$$W_{52} = \begin{array}{c|ccc|c} & 1 & 4 & 5 & \\ \hline 1 & \infty & \infty & 1 & 1 \\ 2 & 0 & 1 & \infty & \\ 4 & 0 & \infty & 0 & \\ \hline & & 1 & & \end{array}$$

$C_{52}=22+1+1=24$

Оценка для W_{51} наименьшая, значит, переходим к W_{51}^*:

$$W_{51}^* = \begin{array}{c|cc} & 1 & 5 \\ \hline 2 & 0 & \infty \\ 4 & \infty & 0 \end{array}$$

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

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

1→4→~~5~~→3→2

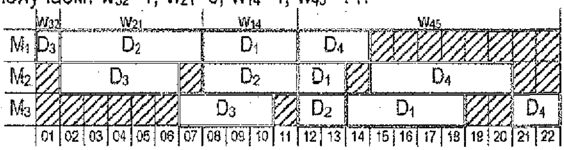
$D_3 \to D_2 \to D_1 \to D_4$ – порядок обработки.

Построим компактное расписание, соответствующее найденному порядку обработки. Обратимся к матрицам P и W_0. По матрице P определяем время обработки каждой детали на каждом станке.

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

$$W_0 = \begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & \infty & 4 & 5 & 4 & 11 \\ 2 & 6 & \infty & 9 & 7 & 12 \\ 3 & 4 & 1 & \infty & 3 & 10 \\ 4 & 5 & 3 & 8 & \infty & 11 \\ 5 & 0 & 0 & 0 & 0 & \infty \end{array}$$

По матрице W_0 получаем: $w_{32}=1$, $w_{21}=6$, $w_{14}=4$, $w_{45}=11$.



Время обработки по полученному расписанию составляет 22 единицы. Задача решена.

__Примечание.__ Приведем пример того, как рассчитываются величины w_{ij}. Найдем w_{23}.

$$w_{23} = \max_{1\le h\le 3}\left\{\sum_{s=1}^{h} p_{s2} - \sum_{s=1}^{h-1} p_{s3}\right\} = \max(\underbrace{p_{12}-0}_{h=1}, \underbrace{(p_{12}+p_{22})-p_{13}}_{h=2}, \underbrace{(p_{12}+p_{22}+p_{32})-(p_{13}+p_{23})}_{h=3}) =$$

$$= \max(6-0, (6+4)-1, (6+4+2)-(1+5)) = \max(6,9,6)=9.$$

Это же значение $w_{23}=9$ можно получить, построив фрагмент диаграммы для $D_2 \to D_3$.

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

Для решения задачи составления РГО реализуем следующую *математическую модель.*

<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>

$$z=\sum_{i=1}^{n}\sum_{\substack{k=1\\ i\neq k}}^{n} w_{ik}\cdot x_{ik}\to\min;$$ – общее время прохождения цикла.

<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;

$$\begin{cases} u_i-u_k+(n-1)\cdot x_{ik}\leq n-2,\\ 2\leq i\neq k\leq n.\end{cases}$$ – (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 матрицу *W*, предварительно полученную в *Derive*.

	A	B	C	D	E	F	G	H	I	J
1	Время от начала обработки детали Di до начала обработки детали Dj на первом станке:									
2			1	2	3	4	5			
3		1	100000	4	5	4	11			
4		2	6	100000	9	7	12			
5	W=	3	4	1	100000	3	10			
6		4	5	3	8	100000	11			
7		5	0	0	0	0	100000			

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

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

	A	B	C	D	E	F	G	H	I
9	Признак того, обрабатывается деталь Dj сразу же после детали Di, или нет:								
10			1	2	3	4	5		
11		1						0	
12		2						0	
13	X=	3						0	
14		4						0	
15		5						0	
16			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 1 (если сразу же за *Di* обрабатывается *Dk*), либо 0 (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:

$\sum\limits_{\substack{k=1 \\ i\neq k}}^{n} x_{ik}=1,\ i=\overline{1,n};$ – 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*-й строке матрицы X должна равняться единице);
формула в **H11**: **=СУММ(C11:G11)**, тиражируем ее на **H11:H15**.

$\sum\limits_{\substack{i=1 \\ i\neq k}}^{n} x_{ik}=1,\ k=\overline{1,n};$ – перед обработкой детали *Dk* должна обрабатываться в точности одна из имеющихся деталей, за исключением детали *Dk* (это условие равносильно тому, что сумма элементов *k*-го столбца матрицы X должна равняться единице);
формула в **C16**: **=СУММ(C11:C15)**, тиражируем ее на **C16:G16**.

3. Введем формулу, задающую выражение для целевой функции Z, определяющую общее время обработки всех деталей. В ячейку **H17**: **=СУММПРОИЗВ(C3:G7;C11:G15)**

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

	A	B	C	D	E	F	G	H	I
17				Общее время обработки всех деталей:				0	
18			Количество n всех деталей (включая фиктивную):					5	
19		Сумма всех элементов главной диагонали матрицы X:						0	

4. Укажем количество всех деталей, включая фиктивную, т.е. зададим размерность матрицы W (в нашем примере вводим в ячейку **H18**: **=5**).

5. Введем формулу, вычисляющую сумму всех элементов главной диагонали матрицы X (необходимо для ограничения, контролирующего равенство нулю всех элементов главной диагонали матрицы X). Вводим в ячейку **H19**: **=C11+D12+E13+F14+G15**

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

	A	B	C	D	E	F	G	H	I	J
21	Числа u_i, участвующие в ограничениях вида $u_i - u_k + (n-1)*x_{ik} <= n-2$, где i, k = 2,...,n:									
22			u2	u3	u4	u5				
23										

7. 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:

	A	B	C	D	E	F	G
25	Ограничения вида: $u_i - u_k + (n-1)*x_{ik} <= n-2$, где i, k = 2,...,n:						
26			u2	u3	u4	u5	
27		u2	0	0	0	0	
28		u3	0	0	0	0	
29		u4	0	0	0	0	
30		u5	0	0	0	0	

Например, выражению

$$u_5-u_2+(n-1)\cdot x_{52}$$

соответствует формула

=$F$23-C$23+($H$18-1)*D15,

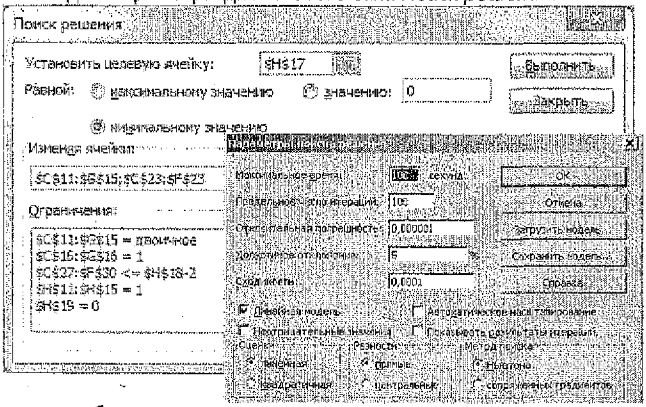
содержащаяся в ячейке **C30**.

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

Окончательно в формульном виде получим таблицу следующего вида:

	A	B	C	D	E	F
26			u2	u3	u4	u5
27		u2	=$C$23-C23+($H$18-1)*D12	=$C$23-D23+($H$18-1)*E12	=$C$23-E23+($H$18-1)*F12	=$C$23-F23+($H$18-1)*G12
28		u3	=$D$23-C23+($H$18-1)*D13	=$D$23-D23+($H$18-1)*E13	=$D$23-E23+($H$18-1)*F13	=$D$23-F23+($H$18-1)*G13
29		u4	=$E$23-C23+($H$18-1)*D14	=$E$23-D23+($H$18-1)*E14	=$E$23-E23+($H$18-1)*F14	=$E$23-F23+($H$18-1)*G14
30		u5	=$F$23-C23+($H$18-1)*D15	=$F$23-D23+($H$18-1)*E15	=$F$23-E23+($H$18-1)*F15	=$F$23-F23+($H$18-1)*G15

8. Введем параметры диалогового окна *Поиск решения:*



– укажем целевую ячейку, содержащую общее время обработки всех деталей;
– установим ее равной минимальному значению;
– наши 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 ф-лам $u_i - u_k + (n-1)\cdot x_{ik}$, должны быть $\leq n$-2.

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

а) 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 – 22 единицы)

	A	B	C	D	E	F	G	H	I
9	Признак того, обрабатывается деталь Dj сразу же после детали Di, или нет:								
10			1	2	3	4	5		
11		1	0	0	0	1	8,327E-17	1	
12		2	1	0	0	0	0	1	
13	X=	3	0	1	0	8,327E-17	0	1	
14		4	0	8,327E-17	0	0	1	1	
15		5	0	0	1	0	0	1	
16			1	1	1	1	1		
17				Общее время обработки всех деталей:				22	

б) таблица со значениями u_i (они являются вспомогательными).

По значениям матрицы X, равным 1, определим порядок, в котором должны обрабатываться детали за минимальное время. Формирование решения начнем с фиктивной детали. Видим, что x_{53}=1, значит, сразу после детали *D5* должна обрабатываться деталь *D3*. Аналогично, сразу после *D3* должна обрабатываться *D2*, после *D2 – D1*, после *D1 – D4*, после *D4 – D5*. Таким образом, получен искомый цикл: 5 – 3 – 2 – 1 – 4 – 5. Удалив из него фиктивную 5-ю деталь, получим порядок обработки: *D3 – D2 – D1 – D4*.

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).

Сетевая модель технологического процесса обработки представлена матрицей C. 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 C 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».
5. Указать оборудование, которое оказалось загружено наибольшим количеством заготовок а) в решении, полученном вручную, б) в решении, полученном в Excel.

3.2. 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) есть одна и только одна вершина v_1, называемая источником, такая, что ни одна дуга не заходит в v_1;

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

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

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

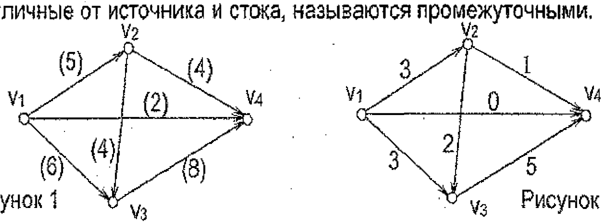


Рисунок 1 Рисунок 2

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: $0 \le \varphi(r) \le c(r)$;
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 сети G называется величина $\bar{\varphi}$, 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, т.е. 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 $\bar{\varphi}$ принимает максимальное значение по сравнению с другими допустимыми потоками в 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. $\bar{\varphi}=6$.

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 $\bar{\varphi}$, 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 $\bar{\varphi}$ 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).

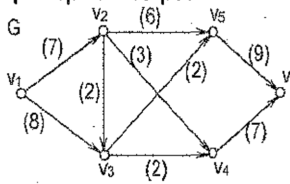


Рисунок 3

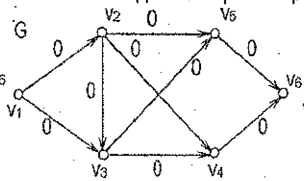


Рисунок 4

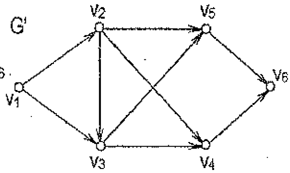


Рисунок 5

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).

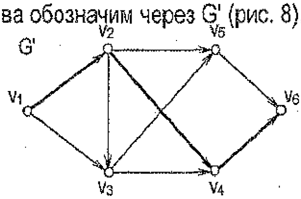


Рисунок 6

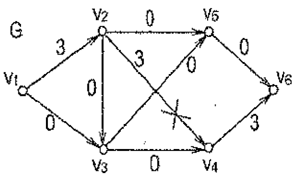


Рисунок 7

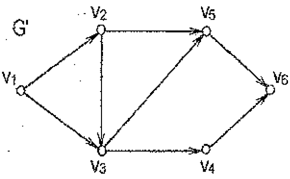


Рисунок 8

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 (v_2,v_3),(v_3,v_4)) (рис.9). Получим поток $\varphi=\varphi_2$, содержащий 3 насыщенные дуги (рис. 10). Удалим полученные насыщенные дуги из G', оставшийся орграф снова обозначим G' (рис. 11).

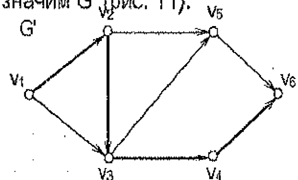


Рисунок 9

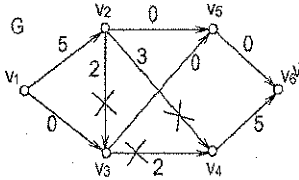
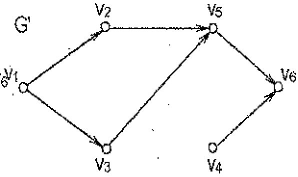


Рисунок 10



Риссунок11

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).

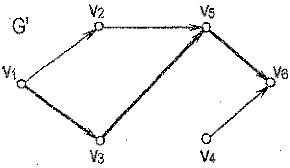


Рисунок 12

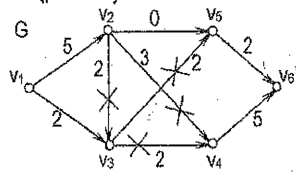


Рисунок 13

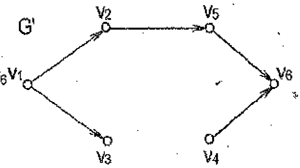


Рисунок 14

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).

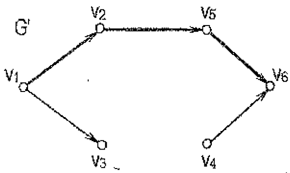


Рисунок 15

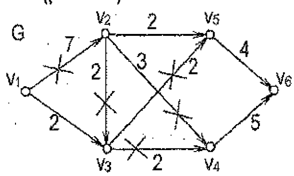


Рисунок 16

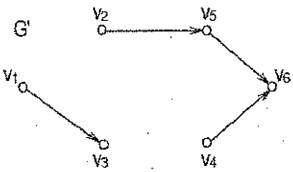


Рисунок 17

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).

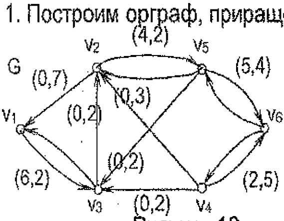


Рисунок 18

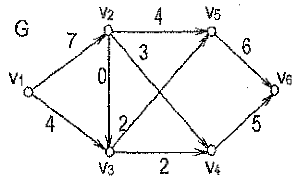


Рисунок 19

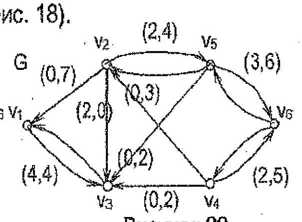


Рисунок 20

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 величина этого потока $\overline{\varphi}_5=11$.

3.3. 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

Транспортная сеть задана матрицей.
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.

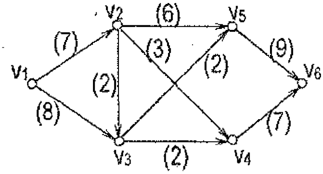


Рисунок 1 – Транспортная сеть

1. Пускаем в сети нулевой поток.
Ищем путь ω_1 из v_1 в v_6 и увеличиваем поток по дугам из ω_1 на величину a.
Далее ищем другой путь из v_1 в v_6 и т.д. 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.

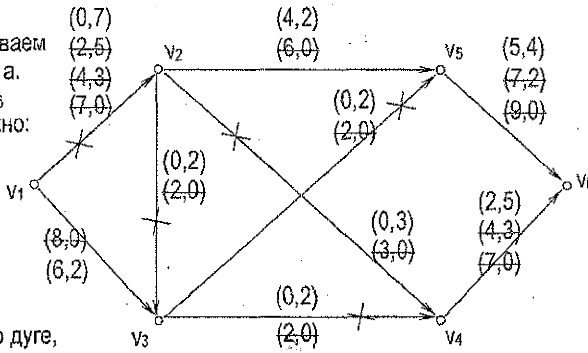


Рисунок 2 – Отыскание полного потока

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:

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

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).

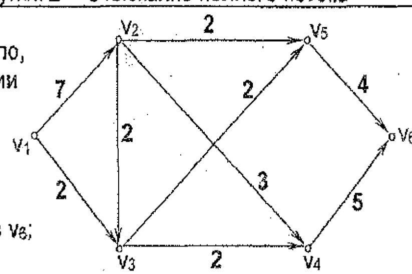


Рисунок 3 – Найденный полный поток

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.

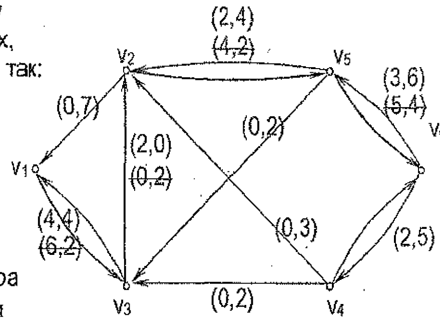


Рисунок 4 – Граф приращений

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:

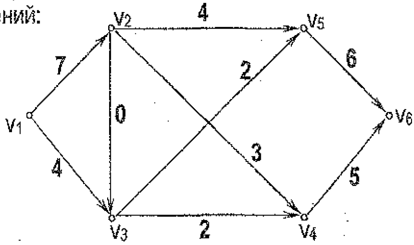


Рисунок 5 – Новый полный поток

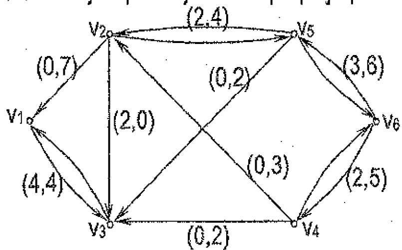


Рисунок 6 – Новый граф приращений

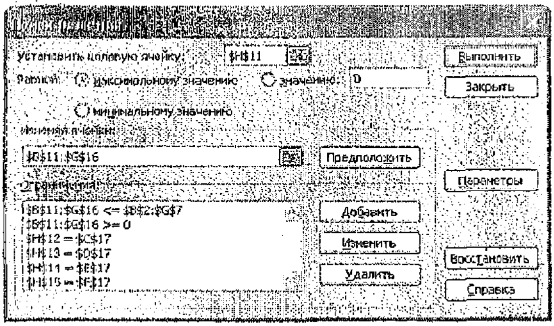
Дальнейшее увеличение потока невозможно – в новом графе приращений нет путей из v_1 в v_6: из источника v_1 можно выйти только в вершину v_3, а из v_3 ни в какую другую вершину движение невозможно, т.к. из v_3 не исходит ни одна дуга.

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: $\bar{\varphi}_{max} = 3+2+2+2+2 = 7+4 = 6+5 = 11$.

3.4. 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: B11:G16, в них разместится найденный поток. Задаем ограничения:



$B$11:$G$16 <= $B$2:$G$7 – для каждой дуги пущенный по ней поток не должен превышать ее пропускной способности;
$B$11:$G$16 >= 0 – поток по каждой дуге должен быть ≥ 0;
$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. Дискретная математика для инженера / О.П. Кузнецов, Г.М. Адельсон-Вельский. – М.: Энергия, 1987.
2. Капустин, Н.М. Автоматизация машиностроения / Н.М. Капустин, Н.П. Дьяконова, П.М. Кузнецов. – М.: Высшая школа, 2003. – 223 с.
3. Нефедов, В.Н. Курс дискретной математики / В.Н. Нефедов, В.А. Осипова. – 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. – 144 с.

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 вёрстка: Тузик И.В.
Корректор: Никитчик Е.В.

Подписано в печать 20.12.2016 г. Формат 60x84 $^1/_{16}$. Бумага «Performer».
Гарнитура «Arial Narrow». Усл. печ. л. 1,86. Уч. изд. л. 2,0. Заказ № 199. Тираж 70 экз.
Отпечатано на ризографе учреждения образования «Брестский государственный технический университет». 224017, г. Брест, ул. Московская, 267.