Частичные автоматы.

Конечный автомат $S=(A,Q,V,\delta,\lambda)$ называется *частичным*, если хотя бы одна из функций δ или λ не является всюду определённой.

Если на некотором наборе значение функции переходов или выхода не определено, то в соответствующей клетке таблицы состояний автомата будем использовать *неопределённый символ* «–».

Через $S(\alpha,q)$ будем обозначать выходное слово, полученное в результате работы над словом α неинициального автомата, запущенного из состояния q. Если на некоторых наборах аргументов значение выходной функции не определено, то в соответствующих позициях выходного слова ставим неопределённый символ.

Говорят, что слово α *применимо к состоянию* q_i неинициального частичного автомата, если функция переходов автомата, начавшего работу в состоянии q_i над словом α, может быть не определена лишь после считывания последнего символа слова α.

Пример 1: Пусть $A=\{a,b\}$, $V=\{0,1\}$, $Q=\{q_1,q_2,q_3\}$, таблица состояний:

A \ Q	q_1	q_2	q_3
a	$q_2,1$	$q_3,1$	–
b	–	$–,0$	$q_3,0$

Проверим работу над словом aba этого неинициального частичного автомата, запущенного из состояний q_2 и q_1.

	a	b	a
q_2	q_3	q_3	–
	1	0	–

	a	b	a
q_1	q_2	–	
	1	0	

Слово aba применимо к состоянию автомата q_2 и результатом работы автомата над этим словом является $1\,0\,-$.

Это можно записать так: $S(aba,q_2)=10-$.

Слово aba неприменимо к состоянию автомата q_1, так как на втором шаге работы автомата над словом произошёл переход в неопределённое состояние.

Будем рассматривать слова одинаковой длины в алфавите, содержащем неопределённый символ «–».

Будем говорить, что слово α *покрывает* слово β, если слово β может быть получено из α заменой некоторого множества (может быть, пустого) символов неопределёнными символами.

Пример 2: Пусть $\alpha = 01--1-0-$, $\beta = -----1-0-$.

Здесь слово α покрывает слово β, так как слово β может быть получено из α заменой первых двух символов неопределёнными символами.

В дальнейшем будем говорить только о неинициальных автоматах.

Будем говорить, что *состояние* q' автомата S' *покрывает состояние* q автомата S, если любое слово α, применимое к состоянию q автомата S, будет также применимо к состоянию q' автомата S', причём $S'(\alpha, q')$ будет покрывать $S(\alpha, q)$.

Частичный *автомат* S' *покрывает* автомат S, если для любого состояния q автомата S, найдётся покрывающее его состояние q' автомата S'.

Пример 3: Пусть автоматы S и S' заданы таблицами состояний:

S	1	2	3
a	$2, x$	$1, x$	$3, y$
b	$3, -$	$-, y$	$-$

S'	$1'$	$2'$
a	$1', x$	$2', y$
b	$2', y$	$-$

Здесь автомат S' покрывает автомат S, так как состояние 1' покрывает состояния 1 и 2, а состояние 2' покрывает состояние 3.

Например, запустим над словом aba автомат S из 2 состояния, а автомат S' из состояния 1':

S	a	b	a
2	1	3	3
	x	$-$	y

S'	a	b	a
$1'$	$1'$	$2'$	$2'$
	x	y	y

Видим, что $S'(aba,1')$ покрывает $S(aba,2)$.

Пусть даны слова α и β одинаковой длины в алфавите, содержащем неопределённый символ «–».

Будем говорить, что слова α и β *совместимы*, если найдётся слово γ, покрывающее как слово α, так и слово β.

$$\alpha = 01{-}1{-}{-}01$$

Пример 4: Пусть $\beta = {-}1{-}1{-}10{-}$, тогда γ покрывает как α,

$$\gamma = 01{-}10101$$

так и β, следовательно, α и β - совместимые слова.

Состояния q и q' частичного автомата S называются *совместимыми*, если для любого слова α, применимого к состояниям q и q', слова $S(\alpha,q')$ и $S(\alpha,q)$ будут совместимы.

Множество попарно совместимых состояний автомата называется *группой совместимости* этого автомата.

Группа совместимости называется *максимальной*, если при добавлении в неё любого состояния, она перестаёт быть группой совместимости.

Класс групп совместимости автомата S называется *группировкой*, если любое состояние автомата S попадает хотя бы в одну группу совместимости.

Группировка, состоящая из всех максимальных групп совместимости данного автомата, называется *максимальной группировкой* данного автомата.

Пример 5: Рассмотрим задачу отыскания максимальной группировки для частичного автомата, заданного диаграммой состояний:

$A \backslash Q$	1	2	3	4	5	6
a	2, -	3, x	4, -	5, y	-	-
b	3, x	5, x	6, -	3, x	6, x	-, y
c	-	-	3, -	-	-	4, y
d	4, -	-	-	1,-	-	2, -

<u>Этап I.</u> Нахождение пар совместимых состояний.

На первом этапе строим треугольную таблицу и, по аналогии со всюду определёнными автоматами, помечаем крестами клетки, соответствующие парам несовместимых состояний, тогда незачёркнутые клетки будут соответствовать парам совместимых состояний.

Выпишем треугольную таблицу.

2					
3					
4					
5					
6					
	1	**2**	**3**	**4**	**5**

Если для какой-нибудь пары состояний одному значению входного сигнала соответствуют различные выходные символы, следовательно, эта пара состояний не является совместимой, и в соответствующую клетку мы ставим крест:

2					
3					
4		X			
5					
6	X	X		X	X
	1	**2**	**3**	**4**	**5**

Таким образом, определяются пары состояний, несовместимость которых выявляется при считывании входного слова из одного символа (зачёркнуто крестиками чёрного цвета).

Далее, в незачеркнутые клетки выписываем пары состояний, в которые переходит автомат из состояний, соответствующих этой клетке, при подаче на вход одинаковых входных символов.

Из этого правила записи есть три исключения:

1) не выписываем пары одинаковых состояний;
2) не выписываем пару, соответствующую заполняемой клетке;
3) не выписываем пару, если хотя бы один из переходов происходит в неопределённое состояние.

Получим:

	1	2	3	4	5
2	2,3 3,5				
3	2,4 3,6	3,4 5,6			
4	2,5		4,5 3,6		
5	3,6	5,6		3,6	
6			3,4		

Затем, если внутри какой-нибудь клетки выписана пара, соответствующая зачёркнутой ранее клетке, то клетка с этой парой также зачёркивается (крестиком синего цвета). Зачёркнутые синим клетки теперь будут соответствовать парам состояний, несовместимость которых выявляется при считывании входных слов длины 2:

	1	2	3	4	5
2	2,3 3,5				
3	~~2,4~~ ~~3,6~~	~~3,4~~ ~~5,6~~			
4	2,5		4,5 3,6		
5	3,6	~~5,6~~		3,6	
6			3,4		

Затем, если внутри какой-нибудь клетки выписана пара, соответствующая клетке, зачёркнутой синим цветом, мы её также зачёркиваем (крестиком красного цвета). Зачёркнутые красным клетки теперь будут соответствовать парам состояний, несовместимость которых выявляется при считывании входных слов длины 3:

2	~~2,3~~ ~~3,5~~				
3	~~2,4~~ ~~3,6~~	~~3,4~~ ~~5,6~~			
4	~~2,5~~		4,5 3,6		
5	3,6	~~5,6~~		3,6	
6			3,4		
	1	**2**	**3**	**4**	**5**

Получили, что в каждой незачёркнутой клетке выписаны пары состояний, соответствующих только незачёркнутым клеткам. Значит, новых зачёркнутых клеток не возникнет, каждая незачёркнутая клетка соответствует паре совместимых состояний.

<u>Этап II.</u> Построение максимальной группировки.

Построение максимальной группировки проведём по алгоритму, содержащему количество шагов, равное числу внутренних состояний автомата:

а) на нулевом шаге выписываем все состояния, объединяя их в один класс;

б) на *i* - том шаге просматриваем *i* -ый столбец треугольной таблицы.

Если какой-либо класс внутренних состояний, полученный на предыдущем шаге, не содержит состояния с номером *i*, то этот класс переписываем без изменений.

Если класс содержит *i* - е состояние, но все остальные состояния этого класса совместимы с *i* -тым, то этот класс также переписывается без изменений.

Если класс содержит *i* - е состояние вместе с состояниями, несовместимыми с *i* - м, то этот класс делится на две части: одна получается из этого класса удаление состояния с номером *i*, а другая - удалением из этого класс всех состояний, не совместимых с *i* - ым.

Если на каком-то шаге образуется два класса, один из которых является собственным подмножеством другого, то меньший удаляется из рассмотрения.

В результате работы этого алгоритма получается максимальная группировка.

Применим алгоритм к нашему примеру. Работа алгоритма будет состоять из 6 шагов. Выпишем классы, получающиеся на каждом шаге:

0) {1,2,3,4,5,6}
1) {2,3,4,5,6}, {1,5}
2) {1,5}, {3,4,5,6}, {2}
3) {1,5}, {2}, {3,4,5,6}
4) {1,5}, {2}, {3,5,6}, {3,4,5}
5) {1,5}, {2},{3,4,5}, {3,6},{3,5}.

Но {3,5}$\subseteq${3,4,5}, поэтому {3,5} удаляем. В результате получили максимальную группировку {1,5}, {2},{3,4,5}, {3,6}.

Группировка называется *замкнутой*, если для любой группы совместимости этой группировки, для любого символа входного алфавита a и для любых состояний q и q' из группы совместимости Q_v, либо хотя бы одно из значений $\delta(a,q)$ и $\delta(a,q')$ не определено, либо $\delta(a,q)$ и $\delta(a,q')$ определены и из того, что $\delta(a,q)$ принадлежит некоторой группе совместимости Q_w следует, что и $\delta(a,q')$ также принадлежит Q_w.

Теорема о замкнутости максимальной группировки.

Для любого частичного автомата его максимальная группировка замкнута.

Пусть S – произвольный частичный автомат, а $G=\{Q_1,Q_2,\ldots,Q_m\}$ - его максимальная группировка. Допустим, она не замкнута. Тогда найдутся: максимальная группа совместимости Q_v, некоторый символ a входного алфавита и некоторые состояния q и q' из группы совместимости Q_v, такие, что $\delta(a,q)$ и $\delta(a,q')$ определены, но не принадлежат никакой одной группе совместимости автомата.

Значит, найдётся входное слово α, что $S(a\alpha, q)$ и $S(a\alpha, q')$ - несовместимые слова, а, следовательно, состояния q и q' несовместны, что противоречит тому, что q и q' принадлежат одной группе совместимости Q_v. Таким образом, доказана замкнутость максимальной группировки. ■

Теорема о построении покрывающего автомата на основе замкнутой группировки.

На основе любой замкнутой группировки автомата S можно построить покрывающий его автомат S'.

Пусть $G=\{Q_1, Q_2, \ldots, Q_m\}$- некоторая замкнутая группировка автомата S. Каждой группе совместимости Q_v, где $v=1,\ldots,m$, необходимо проставить в автомате S'состояние $q_v^{'}$.

Состояние $q_w^{'}$ в которое переходит $q_v^{'}$ под действием входного символа *a*, определяется группой Q_w, содержащей все состояния, в которые входной сигнал *a* переводит состояния из Q_v. Если таких групп несколько, то можно взять любую из них.

Поскольку Q_v образует группу совместимости, то подача символа *a* приводит в каждом состоянии из Q_v либо к неопределённому символу на выходе, либо к символу, общему для всей группы Q_v. Этот символ и принимается в качестве выходного значения, соответствующего переходу в автомате S' из состояния $q_v^{'}$ под действием символа *a*. Если же в каждом состоянии из Q_v при подаче *a* возникает лишь неопределённый символ, то значение функции выхода автомата S' может быть назначено произвольно, либо может считаться неопределённым.

Покажем, что автомат S'покрывает автомат S.

Возьмём произвольное $q_v \in Q_v$, которому соответствует состояние $q_v^{'}$ автомата S'. Пусть автоматы S и S' установлены в состояниях q_v и $q_v^{'}$ соответственно, и на их входы подаётся входное слово $a_1 a_2 ... a_p$, применимое к состоянию q_v.

Тогда, если $\lambda(a_1, q_v)$ определено, то это же значение будет принимать и $\lambda'(a_1, q_v^{'})$. Если же $\lambda(a_1, q_v)$ не определено, то $\lambda'(a_1, q_v)$ также не определено или $\lambda'(a_1, q_v)$ принимает любое значение выходного алфавита.

Пусть $\delta(a_1, q_v) = q_j$, $q_j \in Q_w$, тогда $\delta'(a_1, q_v^{'}) = q_w^{'}$.

Продолжим рассуждения относительно автоматов S и S', установленных в состояниях q_j и $q_w^{'}$ соответственно, считывающих входной символ a_2, и т.д.

Таким образом, слово $a_1 a_2 ... a_p$ окажется применимым к состоянию $q_v^{'}$, а соответствующее выходное слово автомата S' будет покрывать выходное слово автомата S. ■

Свойство замкнутости группировки гарантирует возможность построение таблицы состояний автомата S'.

Вернёмся к автомату S из примера 5, для которого мы нашли замкнутую максимальную группировку {1,5}, {2},{3,4,5}, {3,6},{3,5}, и построим на основе этой группировки покрывающий автомат S'.

Введём обозначения:

1' = {1,5}, 2' = {2}, 3' ={3,4,5}, 4' = {3,6}, 5' ={3,5}.

Далее, $\delta(a, 1) = 2$, $\delta(a\;,\;5$ не определено, $2 \in 2'$, поэтому $\delta'(a, 1') = 2'$. $\lambda(a, 1)$ и $\lambda(a, 5)$ не определены, поэтому $\lambda'(a, 1')$

можно считать неопределённым, или, например, можно определить $\lambda'(a,1') = y$, что мы и сделаем.

Затем, $\delta(b,1) = 3$, $\delta(b,5) = 6$, и $\{3,6\} \subseteq 4'$, то положим $\delta'(b,1') = 4'$.

$\lambda(a,1) = \lambda(a,5) = x$, значит, $\lambda'(b,1') = x$.

Продолжая далее аналогичным образом, получим таблицу состояний автомата S':

$A \diagdown Q'$	1′	2′	3′	4′	5′
a	2′, y	4′, x	3′, y	3′, -	3′, -
b	4′, x	3′, x	4′, x	4′, y	4′, x
c	-	-	3′, -	3′, y	4′, x
d	3′, -	-, y	1′, y	2′, -	-

Проверим работу автомата S, запущенного из состояния 6 над словом *daaddb*:

S	d	a	a	d	d	b
6	2	3	4	1	4	3
	-	x	-	-	-	x

Так как $6 \in 4'$, то запустим автомат S' над тем же словом *daaddb* из состояния $4'$:

S'	d	a	a	d	d	b
4′	2′	4′	3′	1′	3′	4′
	-	x	-	y	-	x

Получили, что слово $\alpha = daaddb$, применимое к состоянию 6 автомата S применимо также к состоянию 4′ автомата S', а $S'(\alpha,4') = -x-y-x$ покрывает слово $S(\alpha,6) = -x---x$.

Автомат, покрывающий частичный автомат S и имеющий наименьшее возможное число внутренних состояний среди всех автоматов, покрывающих S, называется *минимальным* для S.

Автомат, полученный на основе максимальной группировки, не обязательно будет минимальным.

Рассмотрим так называемый Т- алгоритм, дающий, как правило, покрывающий автомат, имеющий меньшее число внутренних состояний, чем исходный, а также меньшее число внутренних состояний автомата, построенного на основе максимальной группировки.

Т – алгоритм построения покрывающего автомата.

а) Выявляем все пары совместимых состояний (например, с помощью треугольной таблицы).

б) Выписываем все состояния автомата в порядке возрастания индексов. Вычёркиваем все состояния, несовместимые с первым, выбираем следующее по порядку возрастания индексов не вычеркнутое состояние, зачёркиваем все состояния, несовместимые с ним т.д. В результате получим некоторую группу совместимости.

Составляем из вычеркнутых состояний список по порядку возрастания номеров и проделываем с ним ту же последовательность действий, что была описана выше.

Если мы повторим эту процедуру до тех пор, пока это возможно, то получим группировку.

в) Начинаем строить покрывающий автомат, в котором каждой группе совместимости соответствует некоторое внутреннее состояние. Если группировка замкнута, такое построение возможно, и полученный автомат будет искомым.

Если группировка не замкнута, строим на её основе замкнутую группировку, расширяя, в случае необходимости, группы совместимости, или добавляя новые группы совместимости.

После получения замкнутой группировки строим покрывающий автомат, сопоставляя каждой полученной группе совместимости внутреннее состояние автомата.

Пример 6: Построить с помощью Т-алгоритма покрывающий автомат для частичного автомата S, заданного таблицей состояний:

A \ Q	1	2	3	4	5	6	7	8	9
a	2,–	3,x	—	5,x	—	—	1,–	5,y	–,x
b	—	–,y	1,y	—	6,x	–,x	4,y	—	6,–

а) Строим треугольную таблицу для выявления всех пар совместимых состояний по аналогии с тем, как мы это делали в примере 5:

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

На первом шаге выявлены пары состояний (помечены чёрными крестами), несовместимость которых обнаруживается при подаче входного слова из одного символа.

Продолжаем:

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

Зачёркнутые синим клетки теперь соответствуют парам состояний, несовместимость которых выявляется при считывании входных слов длины 2.

Продолжаем. В клетке, соответствующей паре состояний (3,7) выписана пара 1 4, соответствующая клетка которой на предыдущем шаге получила синий крест. Значит, клетку треугольной таблицы, соответствующую паре (3,7) помечаем красным крестом. Несовместимость пары состояний (3,7) выявляется входным словом длины 3. Получаем:

2	2 3							
3								
4	~~2 5~~	~~3 5~~						
5								
6								
7	1 2	1 3	~~1 4~~	1 5				
8	~~2 5~~						1 5	
9			1 6				4 6	
	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**

Так как все пары состояний, выписанные внутри клеток, соответствуют незачёркнутым клеткам, то процесс зачёркивания новых клеток закончен, каждой незачёркнутой клетке полученной треугольной таблицы соответствует пара совместимых состояний автомата S.

б) Напишем полученную последовательность групп состояний:

1 2 3 4 5 6 7 8 9,   4 5 6 7 8,   7 8.

Итак, найдена группировка: $\{\{1,2,3,9\}, \{4,5,6\}, \{7,8\}\}$

Проверим группировку на замкнутость, пробуя построить автомат S' с входным алфавитом $\{a,b\}$, выходным алфавитом $\{x,y\}$ и множеством внутренних состояний $\{1',2',3'\}$, где $1'=\{1,2,3,9\}$; $2'=\{4,5,6\}$; $3'=\{7,8\}$.

Видим, что при подаче на вход сигнала b исходный автомат переходит из состояний 1,2,3 и 9 либо в неопределённое состояние, либо в 1 или 6. Состояния 1 и 6 вместе не входят ни в какую группу совместимости построенной группировки, и группу 2' нельзя расширить, добавив в неё состояние 1, т.к. оно несовместимо с состоянием 4, вошедшим в 2'.

Итак, добавим новое состояние $4'=\{1,6\}$.

Далее видим, что из состояний 7 и 8 под действием сигнала a исходный автомат переходит в состояния 1 и 5, которые также не вошли ни в одну группу совместимости построенной группировки.

Заметим, что состояния 1, 5 и 6 совместимы, поэтому группа 4'={1,6} может быть расширена добавлением состояния 5, значит, 4' примет вид 4'={1,5,6}.

Заметим, что при подаче сигнала a из состояний 1,5,6 исходный автомат переходит либо в неопределённое состояние, либо во 2 состояние, а при подаче b - либо в неопределённое состояние, либо в состояние 6. Каждое из этих состояний входит в одну из новых групп совместимости. Значит, построенная группировка замкнута и состоит из групп совместимости:

1'={1,2,3,9}; 2'={4,5,6}; 3'={7,8}, 4'={1,5,6}.

Строим таблицу покрывающего автомата S':

A \ Q	**1'**	**2'**	**3'**	**4'**
a	**1'**, x	**2'**, x	**4'**, y	**1'**, -
b	**4'**, y	**2'**, x	**2'**, y	**4'**, x

Проверим работу автоматов S и S' над одним и тем же словом $\alpha = aaabaaa$. Запустим автомат S из состояния 7:

	a	a	a	b	a	a	a
7	1	2	3	1	2	3	-
	-	-	x	y	-	x	-

Так как $7 \in 3'$, запускаем автомат S'из состояния 3':

	a	a	a	b	a	a	a
3'	4'	1'	1'	4'	1'	1'	1'
	y	-	x	y	-	x	x

Видим, что слово α применимо к состоянию 3' автомата S'и слово $S'(\alpha,3') = y - x\,y - x\,x$ покрывает слово $S(\alpha,7) = --x\,y - x-$.

Задание для самостоятельной работы: для автомата S построить покрывающий автомат S'' на базе максимальной группировки, сравнить количество внутренних состояний автоматов S, S' и S''.