

Конечные распознаватели. Основные понятия

Задача распознавания: определить, принадлежит ли заданная цепочка заданному языку.

Язык L — множество цепочек.

X_L — характеристическая функция языка, определена на множестве всех цепочек, составленных из алфавита языка:

если $\alpha \in L$, то $X_L(\alpha) = 1$, иначе $X_L(\alpha) = 0$.

Если функция X_L вычислима (существует алгоритм вычисления значения функции по значению аргумента), то существует машина Тьюринга для её вычисления и её можно использовать для решения задачи распознавания.

Если L — регулярный язык, то вычислить X_L можно *конечным распознавателем* — более простой машиной, чем машина Тьюринга.

Конечный распознаватель определяется пятёркой: $A = \langle S, X, s_0, \delta, F \rangle$,

где

S – конечное непустое множество состояний;

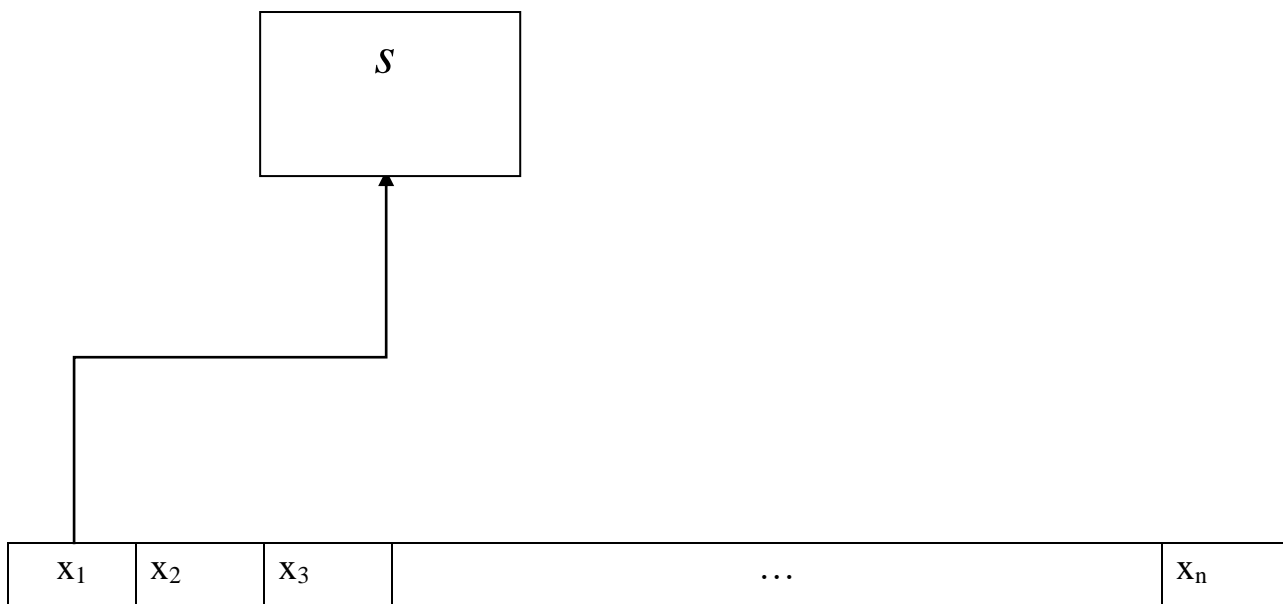
X – конечное непустое множество входных символов;

$s_0 \in S$ – начальное состояние;

$\delta: S \times X \rightarrow S$ – функция переходов, которая паре (состояние, входной символ) ставит в соответствие новое состояние (состояние перехода);

$F \subseteq S$ – множество допускающих состояний.

Конечный распознаватель можно представить как устройство с одним входом, которое в момент времени t может находиться в некотором состоянии $s_t \in S$, а на вход поступает символ $x_t \in X$ из входной цепочки.



Конечный распознаватель определяется пятёркой: $A = \langle S, X, s_0, \delta, F \rangle$,

где

S – конечное непустое множество состояний;

X – конечное непустое множество входных символов;

$s_0 \in S$ – начальное состояние;

$\delta: S \times X \rightarrow S$ – функция переходов, которая паре (состояние, входной символ) ставит в соответствие новое состояние (состояние перехода);

$F \subseteq S$ – множество допускающих состояний.

Пусть

s_t — состояние, в котором находится распознаватель в момент времени t .

x_t — символ, который обрабатывает распознаватель в момент времени t .

Тогда

$s_{t+1} = \delta(s_t, x_t)$ — состояние, в котором будет находиться распознаватель в момент времени $t+1$.

В момент времени $t+1$ распознаватель будет обрабатывать следующий символ x_{t+1} входной цепочки.

В начальный момент времени $t=0$ распознаватель находится в начальном состоянии s_0 и обрабатывает первый символ входной цепочки. На каждом шаге распознаватель обрабатывает новый (следующий за предыдущим) символ цепочки.

Если после обработки последнего символа входной цепочки распознаватель окажется в допускающем состоянии $s_k \in F$, то входная цепочка *принадлежит языку (цепочка допускается распознавателем)*,
иначе – не принадлежит языку (*цепочка отвергается распознавателем*).

Другими словами, конечный распознаватель допускает входную цепочку, если она переводит его из начального состояния в одно из допускающих (позволяет “связать” начальное состояние с одним из допускающих).

Множество (возможно, бесконечное) всех цепочек, допускаемых конечным распознавателем A , образует язык $L(A)$, допускаемый распознавателем A .

Конечный распознаватель определяется пятёркой: $A = \langle S, X, s_0, \delta, F \rangle$,

где

S – конечное непустое множество состояний;

X – конечное непустое множество входных символов;

$s_0 \in S$ – начальное состояние;

$\delta: S \times X \rightarrow S$ – функция переходов, которая паре (состояние, входной символ) ставит в соответствие новое состояние (состояние перехода);

$F \subseteq S$ – множество допускающих состояний.

Конечный распознаватель можно задать таблицей, строки которой соответствуют входным символам, а столбцы – состояниям. Первый столбец соответствует начальному состоянию, а столбцы, соответствующие допускающим состояниям будем отмечать символом “1”. Если распознаватель из состояния s_j при обработке символа x_i переходит в состояние s_k , то в клетке таблицы на пересечении строки x_i и столбца s_j записывается состояние s_k .

Пример.

					1
	s0	s1	s2	s3	s4
ц	s2	s2	s2	s4	s4
.	s3	s3	s4		
+	s1				
-	s1				

В этом распознавателе:

$S = \{s_0, s_1, s_2, s_3, s_4, \text{Error}\}$ — множество состояний;

$X = \{\text{ц}, ., +, -\}$ — множество входных символов;

s_0 — начальное состояние;

функция переходов δ задана таблицей;

$F = \{s_4\}$ — множество допускающих состояний.

Конечный распознаватель

					1
	s0	s1	s2	s3	s4
ц	s2	s2	s2	s4	s4
.	s3	s3	s4		
+	s1				
-	s1				

можно задать ориентированным графом.

Вершины графа — состояния.

Дуги — переходы. На дугах записываются входные символы.

Если распознаватель из состояния s_j при обработке символа x_i переходит в состояние s_k , то из вершины s_j проводится дуга, отмеченная символом x_i , в вершину s_k .

Вершина, соответствующая состоянию ошибки, и дуги, ведущие в состояние ошибки, в графе не изображаются (подразумеваются).

Начальное состояние отмечается стрелочкой, а допускающие состояния выделяются жирной линией.

Пример.

Граф распознавателя, представленного выше таблицей.

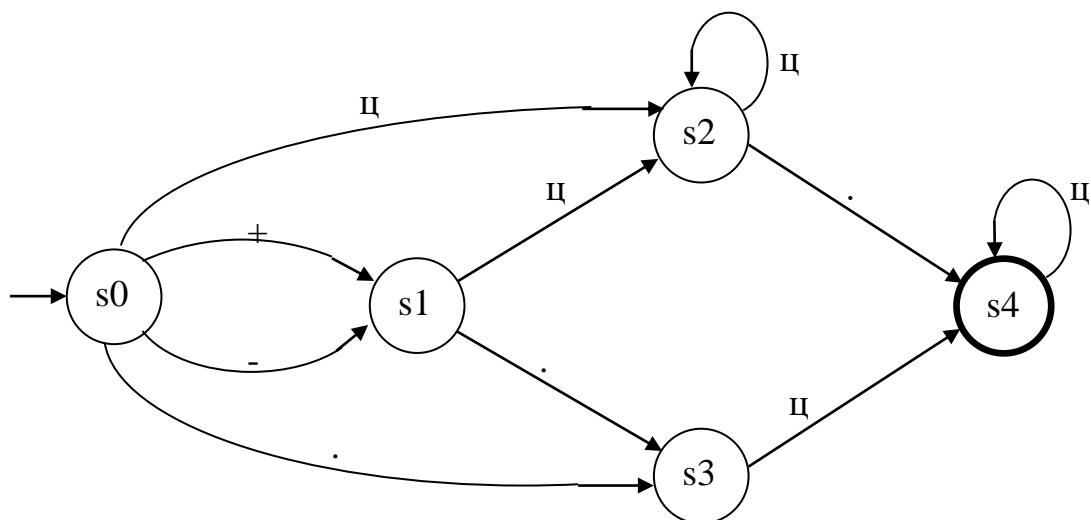
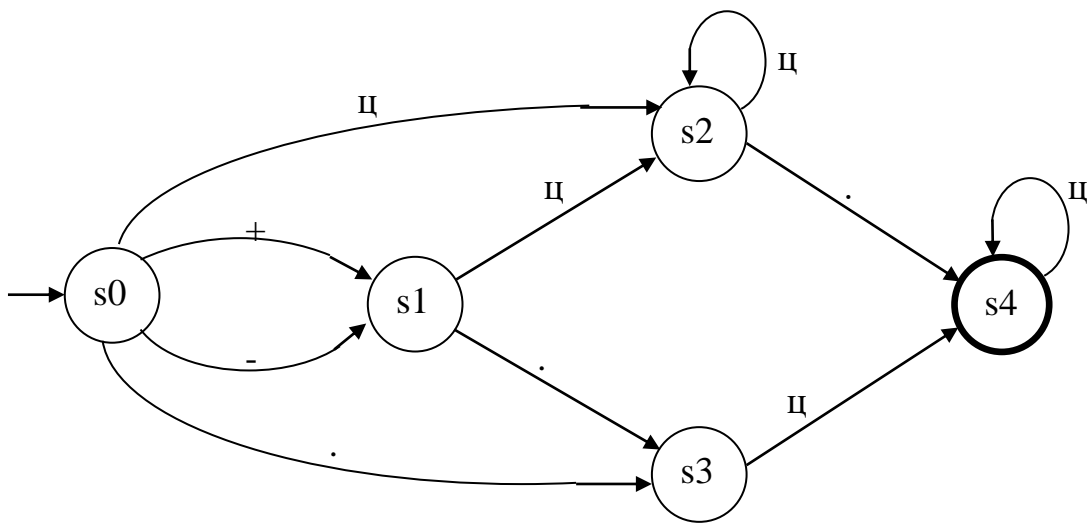


Таблица конечного распознавателя:

					1
	s0	s1	s2	s3	s4
ц	s2	s2	s2	s4	s4
.	s3	s3	s4		
+	s1				
-	s1				

Граф конечного распознавателя:



Цепочка	1	8	.	9	
Состояние	s0	s2	s2	s4	s4

 допустить

Цепочка	1				
Состояние	s0	s2			

 отвергнуть

Цепочка	1	.			
Состояние	s0	s2	s4		

 допустить

Цепочка	—	.	2	2	
Состояние	s0	s1	s3	s4	s4

 допустить

Конечный распознаватель называется *полностью определённым*, если определено состояние перехода для каждой пары (состояние, входной символ). В противном случае распознаватель называется *неполностью определённым* или *частичным*. В дальнейшем будем рассматривать только полностью определённые распознаватели.

Конечный распознаватель, который не может одновременно находиться более чем в одном состоянии, называется *детерминированным*. Определённый выше конечный распознаватель является детерминированным, т.к. имеет только одно начальное состояние (находится в одном состоянии в начальный момент времени) и функция переходов задаёт единственное следующее состояние (состояние перехода) для любой пары (состояние, входной символ).

Конечный распознаватель, который может одновременно находиться более чем в одном состоянии, называется *недетерминированным*. Недетерминированный распознаватель может иметь несколько начальных состояний и может перейти из состояния при обработке входного символа более чем в одно состояние. Недетерминированный конечный распознаватель не подпадает под данное ранее определение конечного распознавателя.

Конечный недетерминированный распознаватель определяется пятёркой: $A = \langle S, X, S_0, \delta, F \rangle$, где

S – конечное непустое множество состояний;

X – конечное непустое множество входных символов;

$S_0 \subseteq S$ – множество начальных состояний;

$\delta: S \times X \rightarrow 2^S$, где 2^S обозначает булеан S , т.е. множество всех подмножеств множества S – функция переходов, которая паре (состояние, входной символ) ставит в соответствие подмножество состояний;

$F \subseteq S$ – множество допускающих состояний.

Недетерминированный распознаватель, так же, как и детерминированный, можно задать таблицей или графом. В клетках таблицы недетерминированного распознавателя записываются множества состояний переходов, а столбцы, соответствующие начальным состояниям, отмечаются символом “↓”. В графе недетерминированного распознавателя из одной вершины могут выходить различные дуги, отмеченные одним и тем же входным символом.

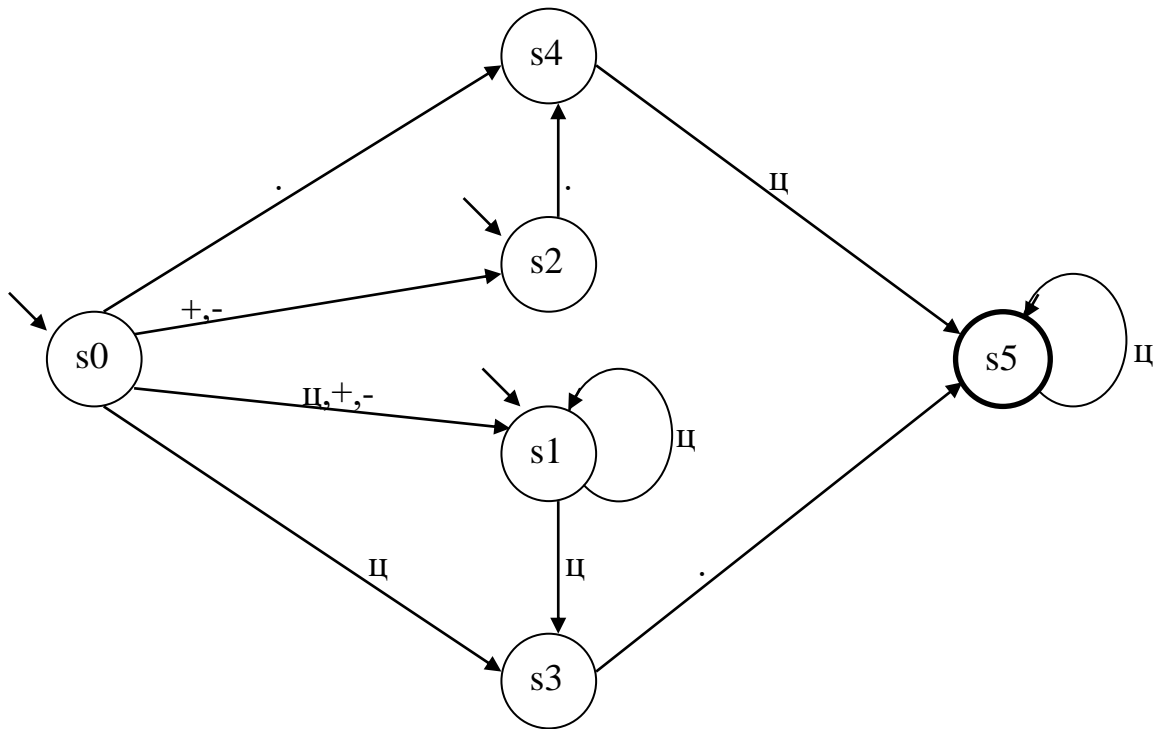
Пример.

Таблица недетерминированного распознавателя.

	↓	↓	↓			1
	s0	s1	s2	s3	s4	s5
ц	s1,s3	s1,s3			s5	s5
.	s4		s4	s5		
+	s1,s2					
-	s1,s2					

Пример.

Граф недетерминированного распознавателя.



Недетерминированный распознаватель может из состояния перейти в пустое множество состояний. Если при обработке входной цепочки множество текущих состояний в некоторый момент времени окажется пустым, то распознаватель прекращает работу и цепочка отвергается.

Цепочка *допускается* недетерминированным конечным распознавателем, если после обработки последнего символа входной цепочки он будет находиться во множестве состояний, содержащих в себе хотя бы одно из допускающих.

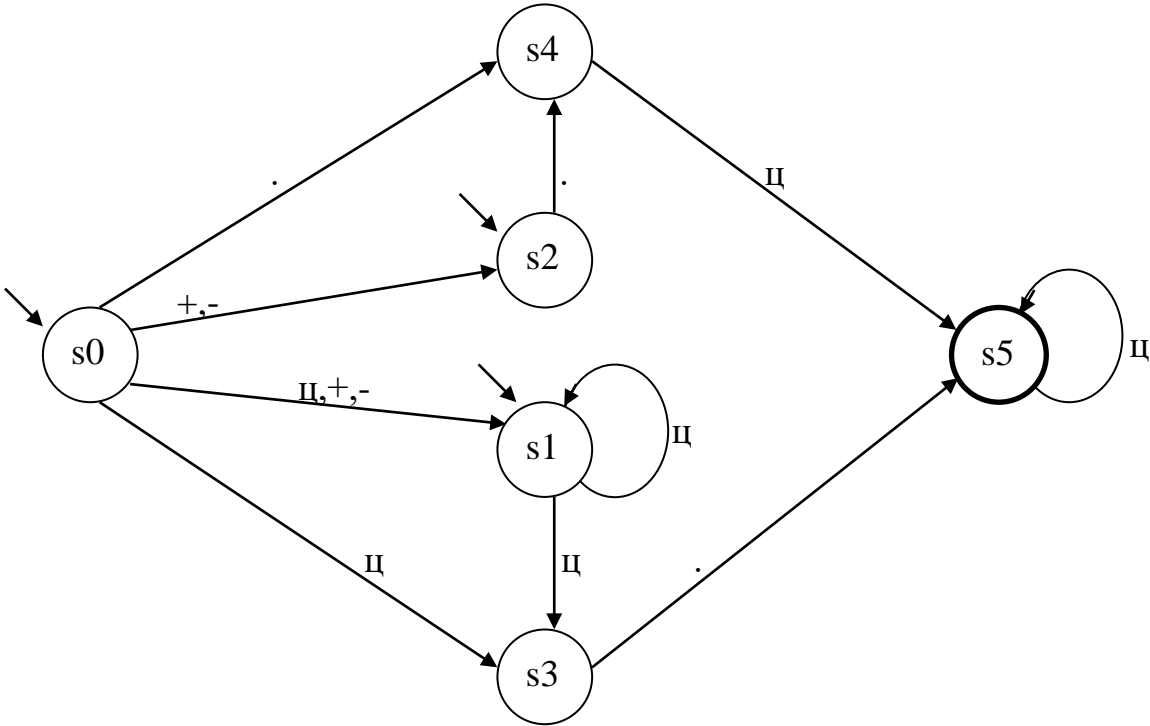
Если распознаватель представить в виде графа, то допустимой цепочке соответствует путь, помеченный символами этой цепочки, из начального состояния в одно из допускающих.

Очевидно, что детерминированные конечные распознаватели являются подклассом недетерминированных конечных распознавателей.

Таблица недетерминированного распознавателя:

	↓	↓	↓			1
	s0	s1	s2	s3	s4	s5
ц	s1,s3	s1,s3			s5	s5
.	s4		s4	s5		
+	s1,s2					
-	s1,s2					

Граф недетерминированного распознавателя:



Цепочка	1	8	.	9	
Состояние	s0, s1, s2	s1, s3	s1, s3	s5	s5

допустить

Цепочка	1				
Состояние	s0, s1, s2	s1, s3			

отвергнуть

Цепочка	1	.			
Состояние	s0, s1, s2	s1, s3	s5		

допустить

Цепочка	—	.	2	2	
Состояние	s0, s1, s2	s1, s2	s4	s5	s5

допустить

Недетерминированный распознаватель, который может перейти на некотором шаге из состояния в состояние без обработки символа входной цепочки, называется *недетерминированным конечным распознавателем с ε -переходами*.

Конечный недетерминированный распознаватель с ε -переходами определяется пятёркой: $A = \langle S, X, S_0, \delta, F \rangle$, где

S — конечное непустое множество состояний;

X — конечное непустое множество входных символов;

$S_0 \subseteq S$ — множество начальных состояний;

$\delta: S \times (X \cup \{\varepsilon\}) \rightarrow 2^S$, где 2^S обозначает булеан S , т.е. множество всех подмножеств множества S — функция переходов, которая паре (состояние, входной символ) или паре (состояние, ε) ставит в соответствие подмножество состояний;

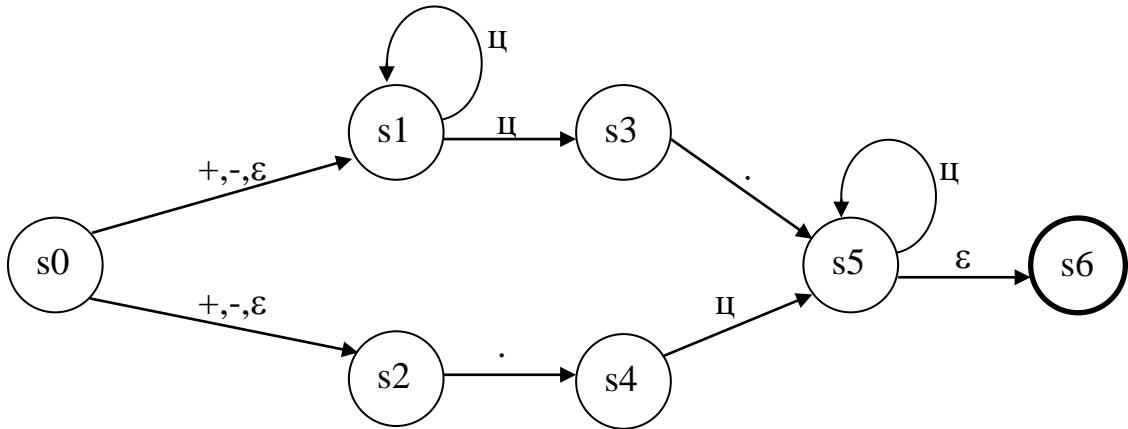
$F \subseteq S$ — множество допускающих состояний.

Таблица недетерминированного распознавателя с ε -переходами отличается от таблицы недетерминированного распознавателя без ε -переходов наличием дополнительной строки, соответствующей пустому символу ε . В графе распознавателя ε -переходу соответствует дуга, отмеченная символом ε .

Таблица недетерминированного распознавателя с ε-переходами:

	↓						1
	s0	s1	s2	s3	s4	s5	s6
ц		s1,s3			s5	s5	
.			s4	s5			
+	s1,s2						
-	s1,s2						
ε	s1,s2					s6	

Граф недетерминированного распознавателя с ε-переходами:



Цепочка	1	8	.	9	
Состояние	s0	s1, s3	s1, s3	s5	s5
	s1, s2			s6	s6

 допустить

Цепочка	1				
Состояние	s0	s1, s3			
	s1, s2				

 отвергнуть

Цепочка	1	.			
Состояние	s0	s1, s3	s5		
	s1, s2		s6		

 допустить

Цепочка	—	.	2	2	
Состояние	s0	s1, s2	s4	s5	s5
	s1, s2			s6	s5

 допустить