Основные понятия и определения

Перязева Юлия Валерьевна

Доцент кафедры ВТ

Теория автоматов

Конечные автоматы с выходом

Основные понятия и определения

Содержание

Детерминированный конечный автомат-преобразователь

Будем считать, что конечный автомат с выходом (КАВ) (автомат-преобразователь) имеет один вход, на который может подаваться за один такт один символ из алфавита входа $A_{BX} = \{a_0, a_1, \ldots, a_n\}$

и один выход, который может принимать в каждый такт одно значение из алфавита выхода $B_{ВЫХ} = \{b_0, b_1, \ldots, b_s\}$.

КАВ может находиться в одном состоянии из конечного множества состояний $Q = \{q_0, q_1, \ldots, q_{r-1}\}$.

Основные понятия и определения

Определение. Конечным автоматом *Мили* с выходом (автоматом-преобразователем) называется система $< A_{BX}, B_{BЫX}, Q, \rho, \lambda >$, где

- функция выхода: $\rho : A_{BX} \times Q \to B_{BЫX}$ – функция выхода, которая указывает что нужно подать на выход, если на вход получен некоторый символ $a_j \in A_{BX}$ и в этот момент КАВ находился в состоянии $q_i \in Q$;
- $\lambda : A_{BX} \times Q \to Q$ – функцию перехода , которая указывает в какое состояние должен перейти КАВ в следующий момент, если на вход получен некоторый символ $a_j \in A_{BX}$ и в этот момент КАВ находился в состоянии $q_i \in Q$;

Основные понятия и определения

Определение. Конечным автоматом *Мура* с выходом (автоматом-преобразователем) называется система $< A_{BX}, B_{\text{ВЫХ}}, Q, \delta, \lambda >$, где

- функция выхода: $\delta : Q \to B_{\text{ВЫХ}}$ – функция выхода, которая зависит только от состояния, в которое перешел автомат;
- $\lambda : A_{BX} \times Q \to Q$ – функцию перехода , которая указывает в какое состояние должен перейти КАВ в следующий момент, если на вход получен некоторый символ $a_j \in A_{BX}$ и в этот момент КАВ находился в состоянии $q_i \in Q$;

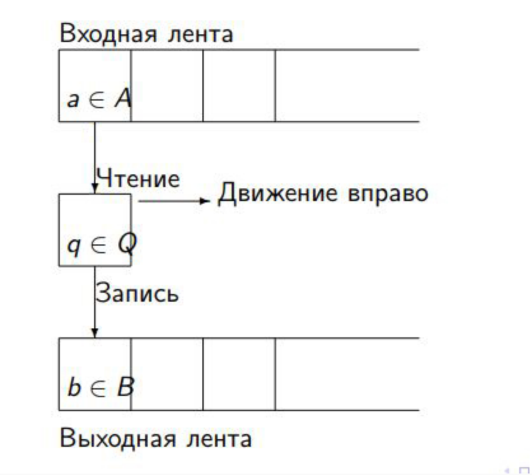
Основные понятия и определения

Если не указано, в каком состоянии находился КАВ в начальный момент времени $t = 0$, то будем говорить, что совокупность $< A_{вх}, B_{вых}, Q, \rho, \lambda >$ задает неинициальный КАВ.

В инициальных же КА $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_0 >$ начальное состояние фиксировано q_0, т. е. они начинают функционировать из одного и того же состояния.

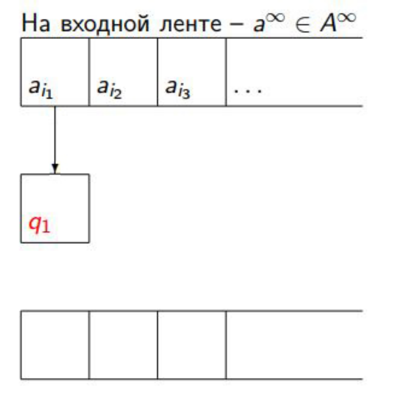
Содержательное понимание КАВ (Мили)

Содержательно КАВ $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_0 >$ можно понимать в виде абстрактного устройства (преобразователя):



Функционирование КАВ (Мили)

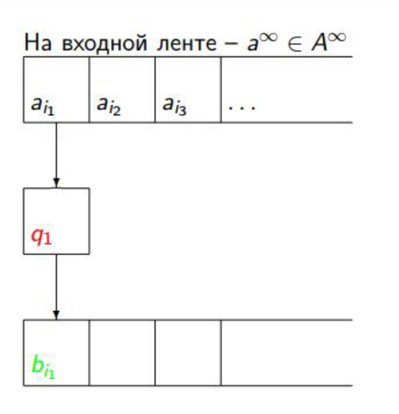
Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_1 >$:



$$t = 1$$
$$b_{i_1} = \rho(ai_1, q_1)$$

Функционирование КАВ (Мили)

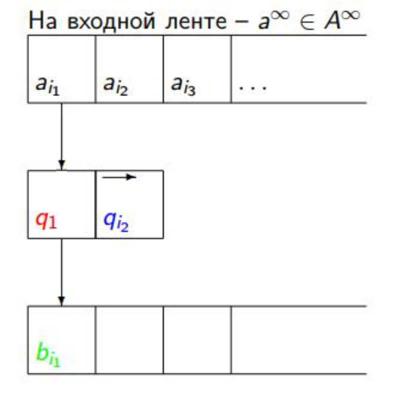
Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_1 >$:



$$t = 1$$
$$b_{i_1} = \rho(ai_1, q_1)$$
$$q_{i_2} = \lambda(ai_1, q_1)$$

Функционирование КАВ (Мили)

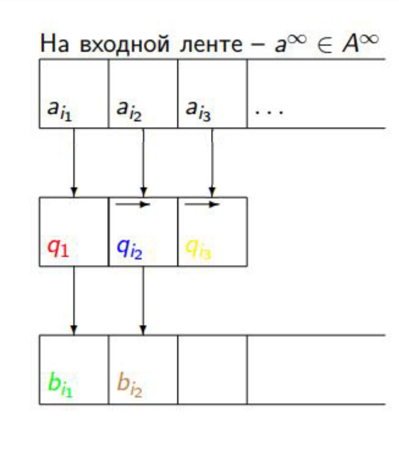
Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_1 >$:



$$t = 1$$
$$b_{i_1} = \rho(ai_1, q_1)$$
$$q_{i_2} = \lambda(ai_1, q_1)$$
$$t = 2$$

Функционирование КАВ (Мили)

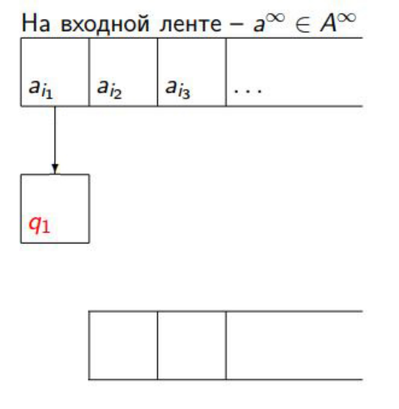
Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \rho, \lambda, q_1 >$:



$$t = 1$$
$$b_{i_1} = \rho(ai_1, q_1)$$
$$q_{i_2} = \lambda(ai_1, q_1)$$
$$t = 2$$
$$b_{i_2} = \rho(ai_2, q_{i_2})$$
$$q_{i_3} = \lambda(ai_2, q_{i_2})$$
$$\ldots$$

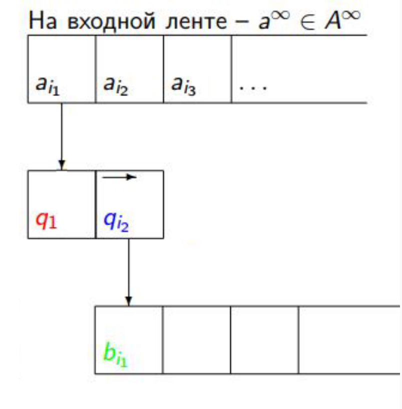
Функционирование КАВ (Мура)

Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \delta, \lambda, q_1 >$:



$$t = 1$$
$$q_{i_2} = \lambda(ai_1, q_1)$$

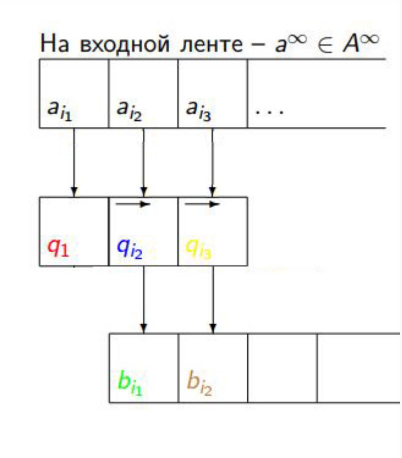
Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \delta, \lambda, q_1 >$:



$$t = 1$$
$$q_{i_2} = \lambda(ai_1, q_1)$$
$$b_{i_1} = \delta(q_{i_2})$$

Функционирование КАВ (Мура)

Фунционирование КАВ $< A_{вх}, B_{вых}, Q, \delta, \lambda, q_1 >$:



$$
\begin{gathered}
t = 1 \\
q_{i_2} = \lambda(ai_1, q_1) \\
b_{i_1} = \delta(q_{i_2}) \\
t = 2 \\
q_{i_3} = \lambda(ai_2, q_{i_2}) \\
b_{i_2} = \delta(q_{i_3}) \\
\ldots
\end{gathered}
$$

Способы задания

Содержание

Конечный автомат. Способы задания

КАВ может быть задан одним из трех способов:

1. **автоматная таблица**, обладает такими качествами как компактность и простота оформления;
2. **диаграмма**, обладает такими качествами, как наглядность и информативность;
3. **канонические уравнения.**

Таблица

Будем говорить, что КАВ (Мили) задан в виде *автоматной таблицы (таблицами)*, если приведена таблица, в которой по горизонтали указаны все внутренние состояния из Q, а по вертикали все символы из $A_{ВХ}$ (или наоборот). На пересечении строки a_j и столбца q_i указывается значения $\rho(a_j, q_i)$ и $\lambda(a_j, q_i)$.

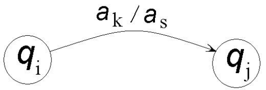
Например, пусть КА задан автоматной таблицей вида:

	q_0	q_1	q_2
а	в / q_1	г / q_1	а / q_2
б	г / q_0	д / q_2	г / q_0
в	д / q_2	а / q_2	в / q_2

$A_{ВХ} = \{а, б, в\}$,
$B_{ВЫХ} = \{а, в, г, д\}$,
$Q = \{q_0, q_1, q_2\}$.

Диаграмма

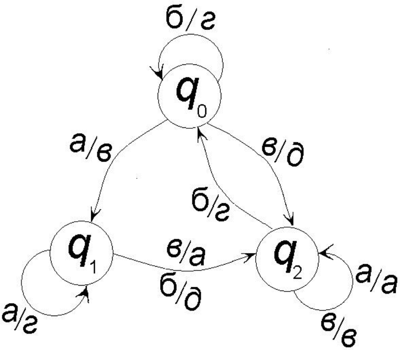
Каждому состоянию КАВ на плоскости ставится в соответствие точка (окружность). Если при подаче на вход символа a_k КАВ переходит из состояния q_i в состояние q_j и на выходе формируется сигнал a_s, то соответствующие точки (окружности) соединяются дугой со стрелкой, рядом с которой помещается указание о входном и выходном сигнале



Способы задания

$A_{вх} = \{а, б, в\}$,
$B_{вых} = \{а, в, г, д\}$,
$Q = \{q_0, q_1, q_2\}$.

	q_0	q_1	q_2
а	в / q_1	г / q_1	а / q_2
б	г / q_0	д / q_2	г / q_0
в	д / q_2	а / q_2	в / q_2



Пример построения КАВ с заданными свойствами

В задачах на построение КАВ необходимо построить диаграмму или автоматную таблицу КАВ, выполняющего определенные, четко сформулированные операции с последовательностями символов из $A_{вх}$.

Пример построения КАВ с заданными свойствами

При рассмотрении примеров работы задаваемого КАВ, будем считать, что сигналы подаются и считываются по очереди справа налево. Например, запись $0312 \to 2110$ обозначает, что

в момент $t = 0$ на вход подается символ 2, КАВ преобразует его

в 0;

в момент $t = 1$ на входе 1, на выходе 1;

в момент $t = 2$ на входе 3, на выходе 1;

в момент $t = 3$ на входе 0, на выходе 2.

Пример построения КАВ с заданными свойствами

$A_{\textit{вх}} = \{0, 1, 2\}$, $B_{\textit{вых}} = \{0, 1, 2, 3, 4\}$. *Построить КАВ, который выдает первый символ входной последовательности без изменения, и далее для каждого поступившего числа – его сумму с предыдущим.*

Пример работы: 1220100212 $\longrightarrow$ 3421102332.

Пример построения КАВ с заданными свойствами

Выделим первоначальное состояние в некоторое особое, в которое больше КАВ в процессе работы не возвращается. Помимо этого, введем еще три состояния: в q_1 автомат переходит, получив 0, в q_2 – получив 1, в q_3 получив 2.

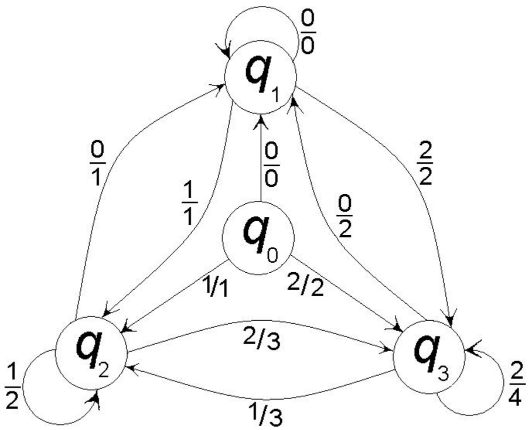
Находясь в q_0 и получив некоторый сигнал $a \in A_{ВХ}$, автомат переходит в q_{a+1}, на выход подается a; находясь в q_{a+1} и получив $b \in A_{ВХ}$, автомат переходит в q_{b+1}, на выход подается $a + b \in B_{ВЫХ}$.

Пример построения КАВ с заданными свойствами

Автоматная таблица имеет вид:

	q_0	q_1	q_2	q_3
0	$0 / q_1$	$0 / q_1$	$1 / q_1$	$2 / q_1$
1	$1 / q_2$	$1 / q_2$	$2 / q_2$	$3 / q_2$
2	$2 / q_3$	$2 / q_3$	$3 / q_3$	$4 / q_3$

Пример построения КАВ с заданными свойствами



Канонические уравнения. Автомат Мили

Детерменированный конечный автомат-преобразователь $< A_{BX}, B_{BЫX}, Q, \rho, \lambda >$ функционирует в дискретные моменты времени $t = 0, 1, 2, \ldots$.

Если рассматривать автомат с n входами $x_1, \ldots, x_n$, на которые могут подаваться символы конечного алфавита A_{BX}, m выходов $y_1, \ldots, y_m$ каждый из которых может принимать значения из конечного алфавита $B_{BЫX}$.

Если обозначить через $x_i(t), y_i(t), q(t)$ значения входа x_i, выхода y_i и состояния q в момент времени t, то работа автомата описывается уравнениями:

$$\left.\begin{array}{l} q(t+1) = \lambda(x_1(t), \ldots, x_n(t), q(t)), \\ y_j(t) = \rho_j(x_1(t), \ldots, x_n(t), q(t))(j = 1, \ldots, m), \end{array}\right\}$$

которые называются *каноническими*.

Канонические уравнения. Пример

Последовательный двоичный сумматор, который может складывать два двоичных числа произвольной разрядности.

Входы x_1 и x_2, один выход y. $A_{ВХ} = B_{ВЫХ} = \{0, 1\}$, и два состояния, соответствующие наличию и отсутствию переноса.

Функционирование начинает с $q(0) = 0$, инициальный автомат. Канонические уравнения последовательного сумматора могут быть записаны в виде:

$$\left.\begin{array}{l} q(t+1) = x_1(t)x_2(t) \vee q(t)(x_1(t) \oplus x_2(t)), \\ y(t) = x_1(t) \oplus x_2(t) \oplus q(t), \\ q(0) = 0 \end{array}\right\}$$

Последовательный двоичный сумматор. Пример

$A_{BX} = \{(00), (01), (10), (11)\}$, $B_{BЫX} = \{0, 1\}$, $Q = \{0, 1\}$
$x(t) = (x_1(t), x_2(t))$

Канонические уравнения:

$$\left.\begin{array}{l} q(t+1) = x_1(t)x_2(t) \vee q(t)(x_1(t) \oplus x_2(t)), \\ y(t) = x_1(t) \oplus x_2(t) \oplus q(t), \\ q(0) = 0. \end{array}\right\}$$

Диаграмма:

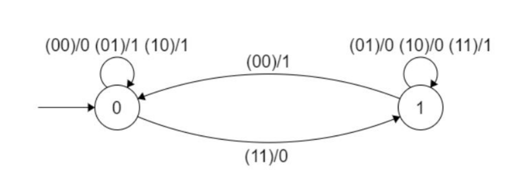


Таблица переходов:

$Q \backslash A_{BX}$	(00)	(01)	(10)	(11)
0	0	0	0	1
1	0	1	1	1

Таблица выходов:

$B_{BЫX} \backslash A_{BX}$	(00)	(01)	(10)	(11)
0	0	1	1	0
1	1	0	0	1

Канонические уравнения. Автомат Мили

В качестве основной модели рассматриваем автоматы с 1 входом x и 1 выходом y.

Конечный автомат $< A_{\text{вх}}, B_{\text{вых}}, Q, \rho, \lambda, q_0 >$ строится по следующим правилам, каноническим уравнениям автомата:

$$\left.\begin{array}{l} q(t+1) = \lambda(x(t), q(t)), \\ y(t) = \rho(x(t), q(t)), \\ q(0) = q_0. \end{array}\right\}$$

Канонические уравнения. Автоматы Мура

В автомате Мура значение выхода однозначно определяется состоянием в тот же момент и не зависит от входного сигнала. Канонические уравнения автомата Мура имеют вид:

$$\left.\begin{array}{l} q(t+1) = \lambda(x(t), q(t)), \\ y(t) = \delta(q(t)). \end{array}\right\}$$

Автоматы Мура. Пример

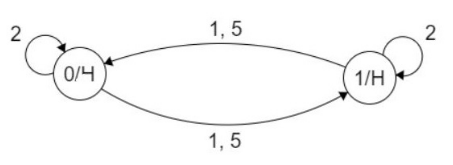
Построить автомат, на вход которого могут поступать монеты 1, 2 и 5 рублей. Автомат выдает сигнал Ч, если сумма опущенных монет четная, и Н, если нечетная.

$A_{ВХ} = \{1, 2, 5\}$, $B_{ВЫХ} = \{Ч, Н\}$, $Q = \{0, 1\}$

Таблица:

$B_{ВЫХ}$	$Q \diagdown A_{ВХ}$	1	2	5
Н	1	0	1	0
Ч	0	1	0	1

Диаграмма:



Канонические уравнения:

$$\left.\begin{array}{l} q(t+1) = (q(t) + x(t))\, mod\, 2, \\ y(t) = \delta(q(t)), \\ q(0) = 0 \end{array}\right\}$$

Автоматное отображение. Информативное дерево

Содержание

Автоматное отображение

В результате работы конечного автомата $\mathfrak{A} =< A_{BX}, B_{BЫX}, Q, \rho, \lambda, q_0 >$ слово $a \in A^\star$ преобразуется в слово $b \in B^\star$, где $A^\star$ множество всех слов алфавита A_{BX}, $B^\star$ множество всех слов алфавита $B_{BЫX}$.

Т.е. конечный автомат А определяет некоторую (словарную) функцию

$$f_{\mathfrak{A}} : A^\star \to B^\star$$

которую будем называть автоматным отображением.

Автоматное отображение

Автоматная функция удовлетворяет следующим условиям:

- сохраняет длину преобразуемого слова;
- осуществляет преобразование совпадающих начальных отрезков входных слов в совпадающие начальные отрезки соответствующих выходных слов.

Автоматное отображение

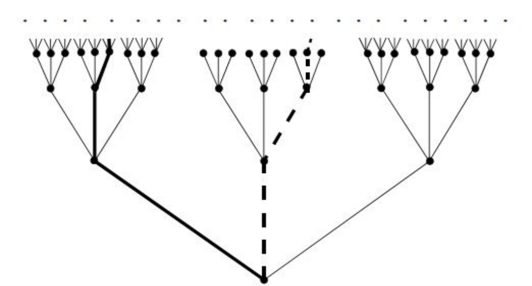
Функция

$$f : A^\star \to B^\star$$

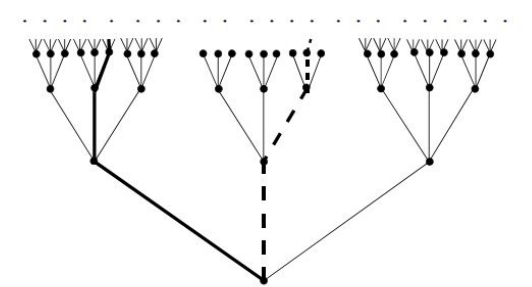
называется автоматной, если найдется такой конечный автомат $\mathfrak{A} =< A_{вх}, B_{вых}, Q, \rho, \lambda, q_0 >$, что $f_{\mathfrak{A}} = f$.

Информативное дерево

С множеством $A^\star$ можно связать некоторое бесконечное дерево T. Пусть $A = \{a_1, a_2, \ldots, a_n\}$. Возьмем любую точку и назовем ее *корнем* дерева. Из корня выпустим n ребер, столько, сколько элементов во входном алфавите, концы назовем вершинами первого яруса. Из каждой вершины первого яруса выпустим n ребер, которые назовем вершинами второго яруса и т.д.



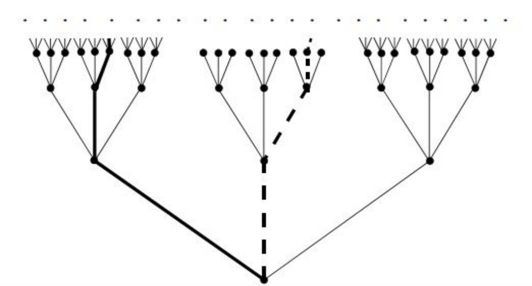
Информативное дерево



Ветви дерева T соответствуют последовательностям $x(1)x(2)\dots x(t)\dots$, причем это соответствие взаимнооднозначное. Будем считать, что ребра, соответствующие буквам алфавита $A_{BX} = \{a_1, a_2, \dots, a_n\}$, идут слева направо (т.е. крайнее левое ребро соответствует букве a_1, следующее – букве a_2, крайнее правое – букве a_n).

Информативное дерево

На рисунке дерево, построенное для для трехэлементного алфавита, например, $A_{BX} = \{a, b, c\}$. Тогда ветвь дерева отмеченная жирной линией будет соответствовать последовательности $abcb\ldots$, а ветвь, отмеченная пунктирной линией, – $bcb\ldots$



Информативное дерево

Пусть конечный автомат $< A_{BX}, B_{BЫX}, Q, \rho, \lambda, q_0 >$ определяет функцию $f : A^\star \to B^\star$. Построим дерево T, соответствующее множеству $A^\star$ и пометим его ребра соответствующими буквами алфавита B.

Рассмотрим последовательность $w = x(1)x(2)\dots x(t)\dots$ которая преобразуется в последовательность $f(w) = y(1)y(2)\dots y(t)\dots$, тогда ребра соответствующей последовательности w ветви дерева пометим символами $y(1), y(2), \dots, y(t), \dots$ Так нужно поступить с каждой ветвью, каждое ребро получит ровно одну отметку.

Информативное дерево

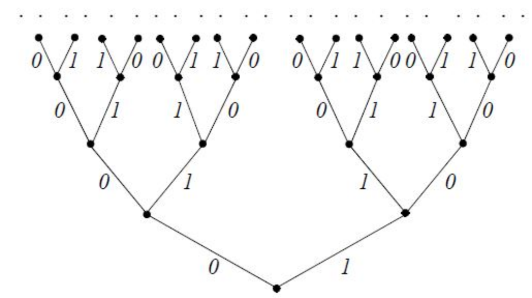
Дерево T, ребра которого помечены вышеописанным способом, назовем **информативным деревом**, соответствующим функции f и обозначим T_f.

Информативное дерево. Пример 1

$A_{BX} = \{0,1\}$, $B_{ВЫХ} = \{0,1\}$,рассмотрим функцию $f : A^{\star} \to B^{\star}$:

$$y(t) = \begin{cases} x(t)\,\text{при}\, t = 1, \\ (x(t) + x(t-1)) mod\, 2\,\text{при}\, t \geq 2; \end{cases}$$

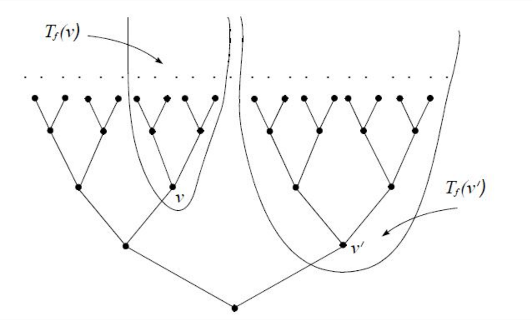
Информативное дерево функции имеет вид:



Информативное дерево

Наоборот, если дано дерево T для $A^\star$, то пометив его ребра буквами из произвольным образом, мы получим информативное дерево некоторой функции f, которая будет однозначно отображать входные слова в выходные. Но для того, чтобы эта функция определяла конечный автомат (конечная память), требуется выполнение еще одного условия.

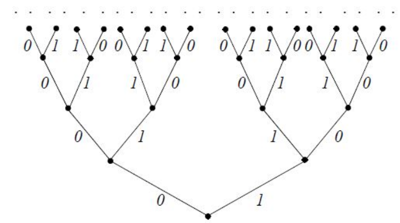
Информативное дерево

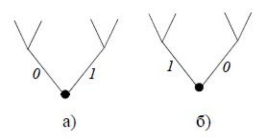


Для любой вершины v дерева обозначим $T_f(v)$ поддерево с корнем в этой вершине. Две вершины эквивалентны $v \sim v^{'}$, если у деревьев $T_f(v)$ и $T_f(v^{'})$ соответствующие друг другу ребра имеют одинаковые пометки. Если множество вершин информативного дерева T_f разбивается на конечное число классов эквивалентности, то тогда f – автоматная функция. Каждому классу эквивалентности в автомате соответствует внутреннее состояние.

Информативное дерево. Пример 1

В дереве каждому ребру поставлен в соответствие один символ, задано взаимнооднозначное соответствие и все вершины в дереве делятся на два вида::





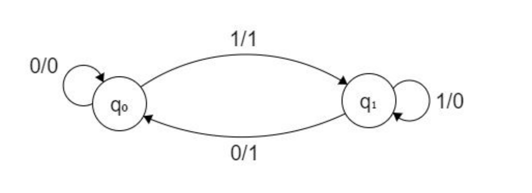
Информативное дерево T_f имеет конечное число классов эквивалентности, два класса, следовательно функция является автоматной и может быть реализована конечным автоматом с двумя состояниями, состояние соответствует классу вершин.

Информативное дерево

$A_{BX} = \{0, 1\}$, $B_{ВЫХ} = \{0, 1\}$, функция $f : A^\star \to B^\star$:

$$y(t) = \begin{cases} x(t) \text{ при } t = 1, \\ (x(t) + x(t-1)) mod\, 2 \text{ при } t \geq 2; \end{cases}$$

Диаграмма автомата:



Функция единичной задержки. Пример 2

$A_{вх} = \{0, 1\}$, $B_{вых} = \{0, 1\}$.

Рассмотрим функцию $f : A^\star \to B^\star$, такую, что последовательность

$$x(1)x(2)\ldots x(t)\ldots$$

преобразуется в последовательность

$$0x(1)x(2)\ldots x(t-1)\ldots$$

.

Такая функция f называется функцией единичной задержки.

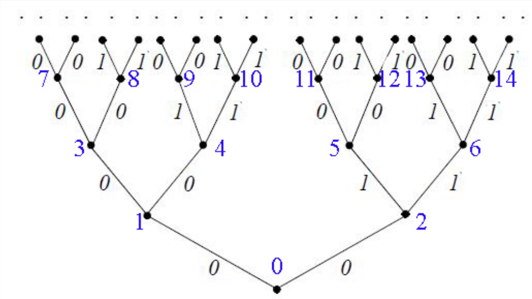
Функция единичной задержки. Пример 2

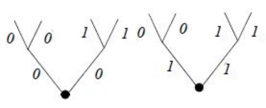
Содержательно, она задерживает элемент входной последовательности на один такт работы и затем записывает его в выходную последовательность. На первом такте работы она выдает 0.

Покажем, что функция f задает автоматное отображение.

Функция единичной задержки. Пример 2

Построим несколько ярусов информативного дерева заданной функции:





Все вершины разбиваются на 2 класса эквивалентности: первый класс вершины $0, 1, 3, 5, \ldots$, второй класс – $2, 4, 6, \ldots$.

Функция единичной задержки. Пример 2

Диаграмма:

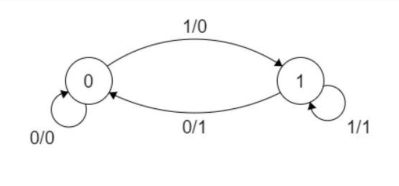


Таблица переходов:

$Q \setminus A_{вх}$	0	1
0	0	1
1	0	1

Таблица выходов:

$Q \setminus A_{вх}$	0	1
0	0	0
1	1	1

Функция единичной задержки. Пример 2

Канонические уравнения:

$$\begin{cases} q(t+1) = x(t), \\ y(t) = \begin{cases} 0 \text{ при } t = 1, \\ x(t) \text{ при } t \geq 2; \end{cases} \\ q(0) = 0. \end{cases}$$

Автоматная функция. Пример 3

Функция задана таблицей, на информативном дереве видно, что взаимнооднозначное отображение и 3 класса эквивалентности: вершина 0 – первый класс, 1 и 2 – второй, 3,4,5,6 – третий класс.

$a_0\ a_1\ a_2$	$b_0\ b_1\ b_2$
0 0 0	0 0 1
0 0 1	0 0 0
0 1 0	0 0 1
0 1 1	0 0 0
1 0 0	0 0 1
1 0 1	0 0 0
1 1 0	0 0 1
1 1 1	0 0 0

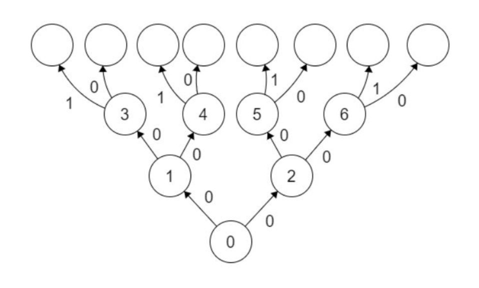
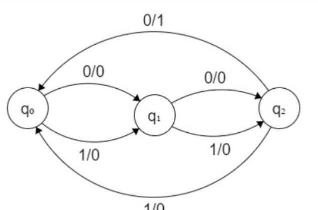


Диаграмма автомата:



Автоматная функция. Пример 3

Можно строить диаграмму автомата по дереву: