

# Дискретная математика 2

Что такое дискретная  
математика?

# Что может быть дискретно?

- Множества, с которыми мы работаем
- Шаги некоторого алгоритма
- Время

# Что такое ДМ?

- Логика высказываний
- Логика предикатов
- Математическая логика
- Комбинаторика
- Теория алгоритмов
- Теория автоматов
- Теория графов
- Теория игр
- Теория кодирования
- Логическое программирование
- Функциональное программирование

# Теория автоматов и формальных языков

Институт Информационных  
Технологий  
ЧелГУ, 2010

A decorative graphic element consisting of several horizontal lines of varying lengths and colors (teal, white, and light blue) extending from the right side of the text area towards the right edge of the slide.

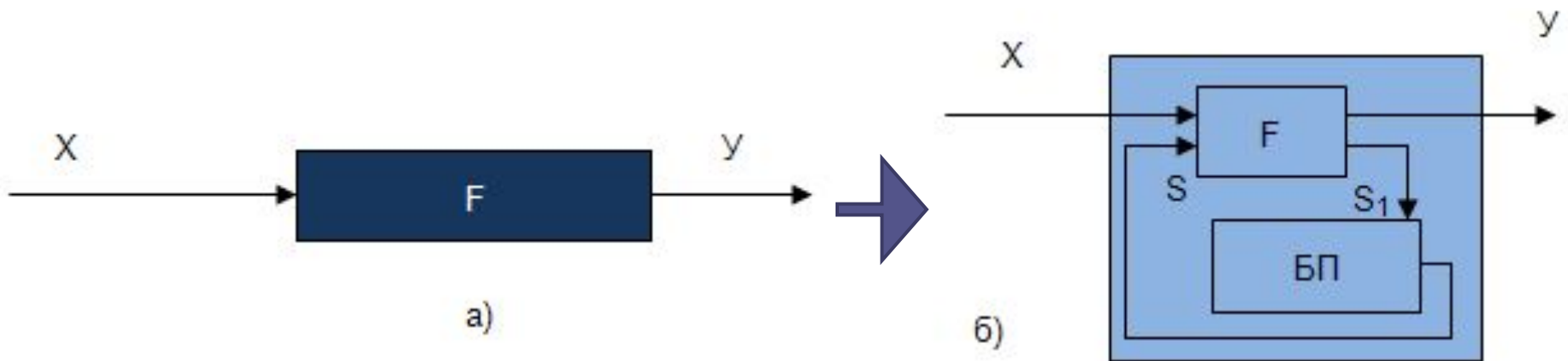
# Автомат



$$F: X \rightarrow Y$$

Зависит от того, какая информация в данный момент появилась на входе  
от того, что происходило раньше

# Автомат

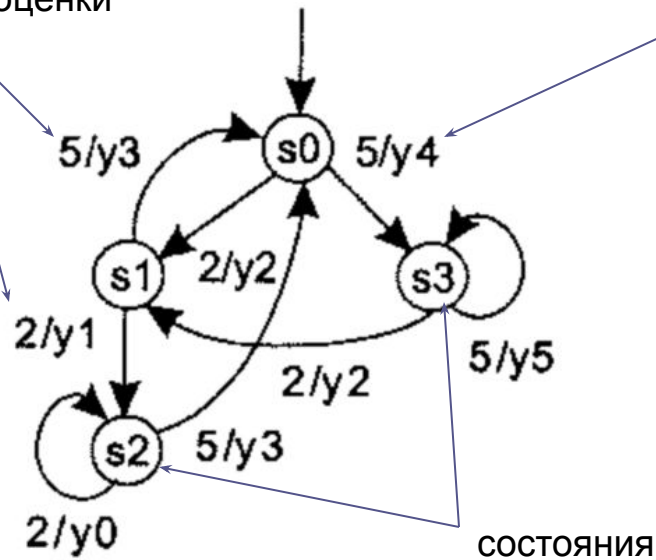


Автомат, в зависимости от входных данных  $X$  меняет свое состояние  $S$  – текущее состояние хранится в памяти

# Автомат

## Пример реализация автомата

Входные данные:  
2,5 - оценки



Действия – выходные сигналы:

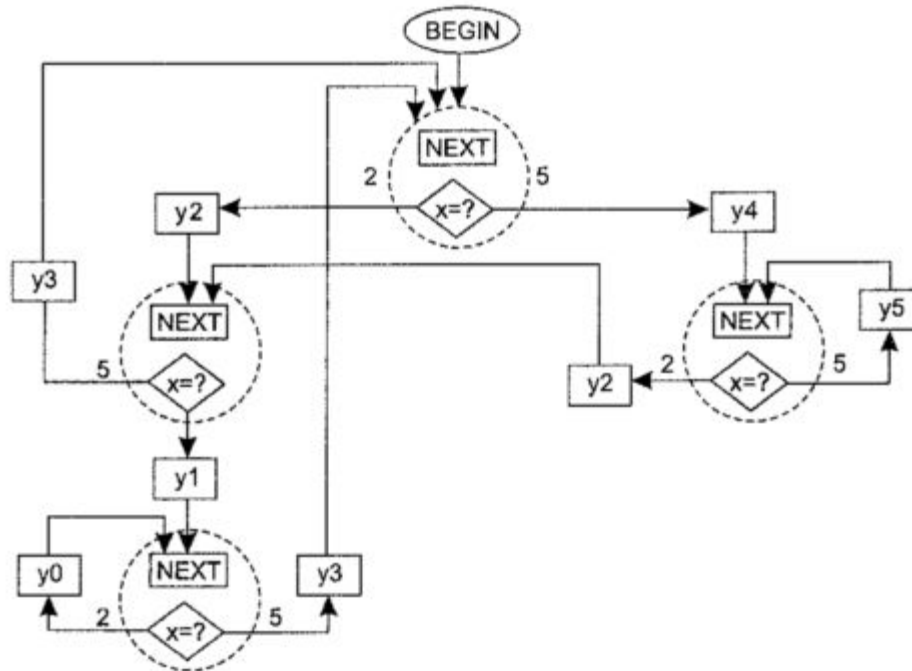
- у0: – брать ремень;
- у1: – ругать сына;
- у2: – успокаивать сына;
- у3: – надеяться;
- у4: – радоваться;
- у5: – ликовать.

Реализация:  
программная  
аппаратная

Автомат, описывающий поведение «умного» отца

# Автомат

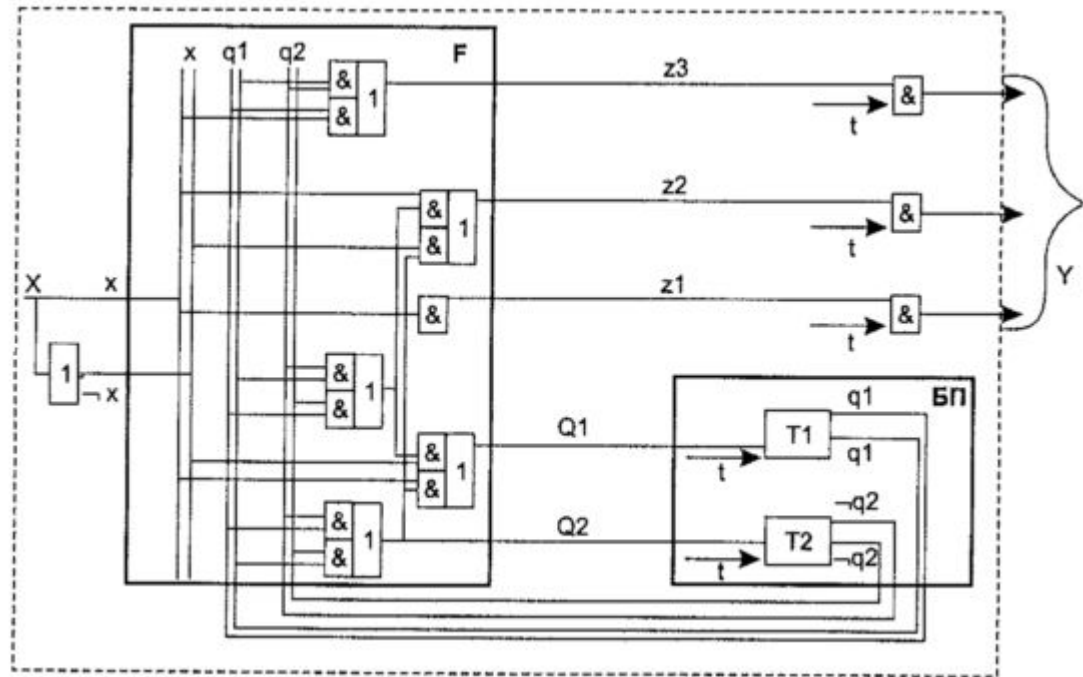
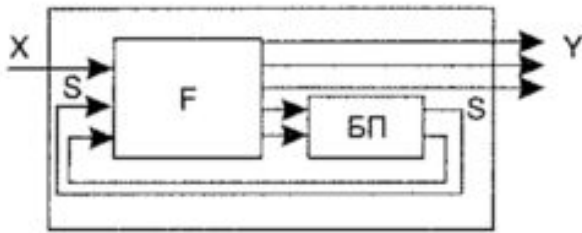
Пример программной реализации автомата





# Автомат

Пример аппаратной реализации автомата



# Автомат

Рассмотрим механизм управления лифтом. Если всего в здании  $N$  этажей, лифт может находиться в одном из  $N$  состояний:

$a_1, a_2, \dots, a_N$  - возможные состояния

На вход подаются номера этажей, к которым должен поехать лифт:

$z_1, z_2, \dots, z_N$  - этажи здания

Выходными сигналами будем считать расстояния, которые должен проехать лифт:

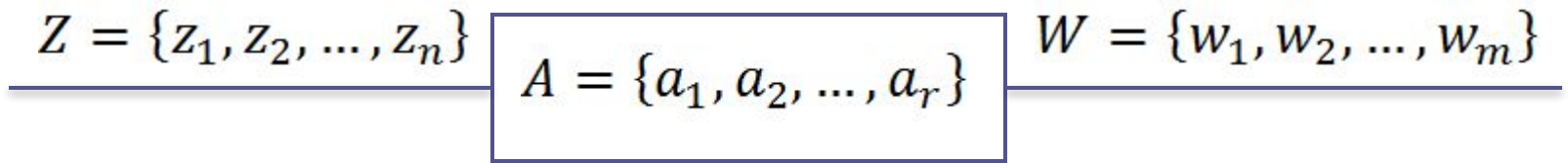
$$w(t) = z(t) - z(t - 1)$$

При этом множество возможных значений  $w(t)$  конечно и определяется набором возможных входных значений и множеством этажей.

# Автомат

- $Z = \{z_1, z_2, \dots, z_n\}$  - входной алфавит
- $W = \{w_1, w_2, \dots, w_m\}$  - выходной алфавит
- $A = \{a_1, a_2, \dots, a_r\}$  - набор внутренних состояний
- $t = 0, 1, 2, \dots$  - дискретные моменты времени
- $a(0)$  - состояние автомата в начальный момент времени
- $a(i)$  - состояние автомата в момент времени  $i$ .

*Автомат* – математическая абстракция, позволяющая описывать пути изменения состояния объекта в зависимости от его текущего состояния и входных данных.



# Алфавитный оператор

Последовательности входных букв:  $z(1)z(2) \dots z(k)$   
называются входными словами.

На вход автомату может подаваться любое слово из *множества допустимы слов*.

Каждое допустимое *входное слово*:  $p = z(1)z(2) \dots z(k)$   
вызывает появление *выходного слова*:  $q = w(1)w(2) \dots w(k)$

Длины соответствующих входных и выходных слов равны между собой.

$\varphi_A: P \rightarrow Q$  - алфавитный оператор, индуцированный автоматом  $A$ .

$P$  - множество допустимых входных слов

$Q$  - множество выходных слов

# Функции переходов и выходов

$\delta$  - функция переходов, если:

$$a(t) = \delta(a(t-1), z(t))$$

$\lambda$  - функция выходов, если:

$$w(t) = \lambda(a(t-1), z(t))$$

При помощи задания начального состояния  $\mathbf{a}(\mathbf{o})$  и функций перехода и выхода  $\lambda$  и  $\delta$  можно для любого входного слова  $\mathbf{p}$  определить выходное слово  $\mathbf{q}$ .

$$\mathbf{q} = \varphi_A(\mathbf{p})$$

# Автоматы Мили и Мура

Автомат Мили:

$$a(t) = \delta(a(t-1), z(t))$$

$$w(t) = \lambda(a(t-1), z(t))$$

Символ на выходе зависит от символа на входе автомата и состояния автомата в предыдущий момент времени

Автомат Мура:

$$a(t) = \delta(a(t-1), z(t))$$

$$w(t) = \lambda(a(t))$$

Символ на выходе зависит только от текущего состояния автомата

Автомат Мура всегда сводится к автомату Мили:

$$\begin{aligned} w(t) &= \lambda(a(t)) = \lambda(\delta(a(t-1), z(t))) \\ &= \tilde{\lambda}(a(t-1), z(t)) \end{aligned}$$

# Автомат Мили

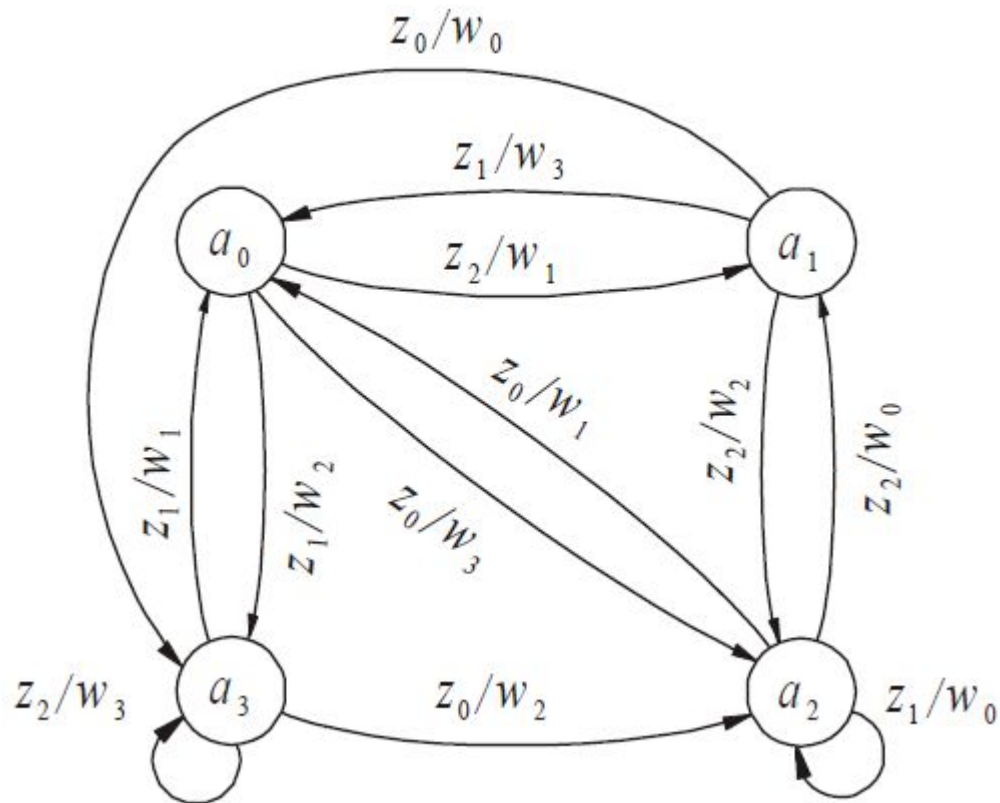
$a(t-1)$	$z_0$	$z_1$	$z_2$
$a_0$	$a_2$	$a_3$	$a_1$
$a_1$	$a_3$	$a_0$	$a_2$
$a_2$	$a_0$	$a_1$	$a_1$
$a_3$	$a_2$	$a_2$	$a_3$

$a(t-1)$	$z_0$	$z_1$	$z_2$
$a_0$	$w_3$	$w_2$	$w_1$
$a_1$	$w_0$	$w_3$	$w_0$
$a_2$	$w_1$	$w_0$	$w_2$
$a_3$	$w_2$	$w_1$	$w_3$

$a(t-1)$	$z_0$	$z_1$	$z_2$
$a_0$	$a_2 / w_3$	$a_3 / w_2$	$a_1 / w_1$
$a_1$	$a_3 / w_0$	$a_0 / w_3$	$a_2 / w_0$
$a_2$	$a_0 / w_1$	$a_1 / w_0$	$a_1 / w_2$
$a_3$	$a_2 / w_2$	$a_2 / w_1$	$a_3 / w_3$

# Автомат Мили

$a(t-1)$	$z_0$	$z_1$	$z_2$
$a_0$	$a_2 / w_3$	$a_3 / w_2$	$a_1 / w_1$
$a_1$	$a_3 / w_0$	$a_0 / w_3$	$a_2 / w_0$
$a_2$	$a_0 / w_1$	$a_1 / w_0$	$a_1 / w_2$
$a_3$	$a_2 / w_2$	$a_2 / w_1$	$a_3 / w_3$





# Автомат Мура

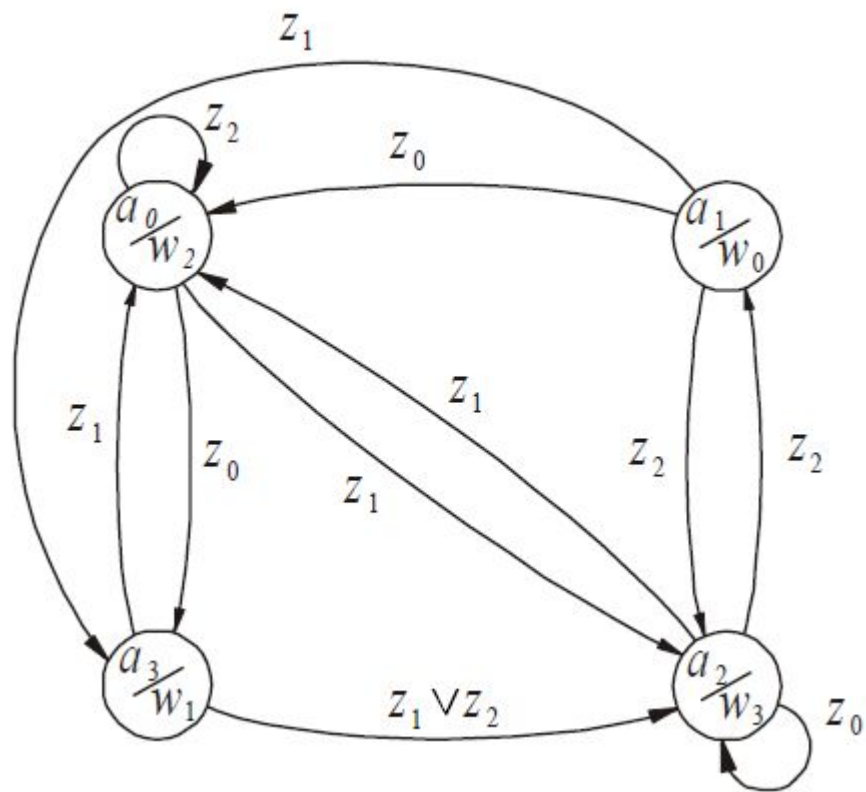
$a(t-1)$	$z_0$	$z_1$	$z_2$
$a_0$	$a_3$	$a_2$	$a_0$
$a_1$	$a_0$	$a_3$	$a_2$
$a_2$	$a_2$	$a_0$	$a_1$
$a_3$	$a_0$	$a_2$	$a_2$

$a(t)$	$w(t)$
$a_0$	$w_2$
$a_1$	$w_0$
$a_2$	$w_3$
$a_3$	$w_1$

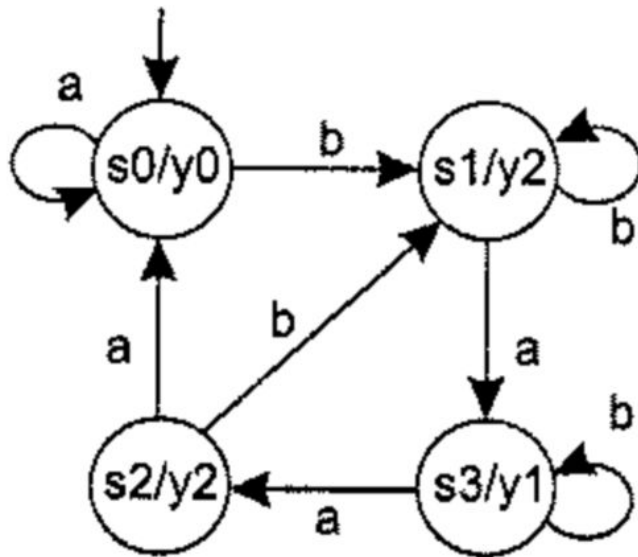
$w(t-1)$	$a(t-1)$	$z_0$	$z_1$	$z_2$
$w_2$	$a_0$	$a_3$	$a_2$	$a_0$
$w_0$	$a_1$	$a_0$	$a_3$	$a_2$
$w_3$	$a_2$	$a_2$	$a_0$	$a_1$
$w_1$	$a_3$	$a_0$	$a_2$	$a_2$

# Автомат Мура

$w(t-1)$	$a(t-1)$	$z_0$	$z_1$	$z_2$
$w_2$	$a_0$	$a_3$	$a_2$	$a_0$
$w_0$	$a_1$	$a_0$	$a_3$	$a_2$
$w_3$	$a_2$	$a_2$	$a_0$	$a_1$
$w_1$	$a_3$	$a_0$	$a_2$	$a_2$

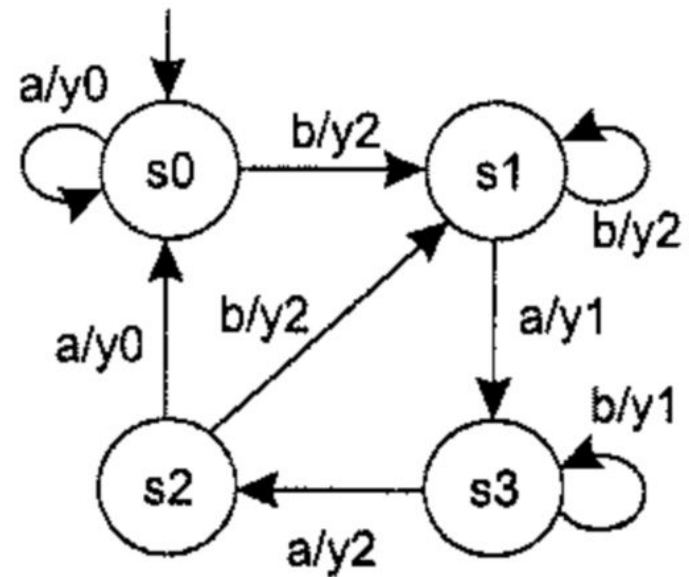


# Автоматы Мили и Мура



Автомат Мура

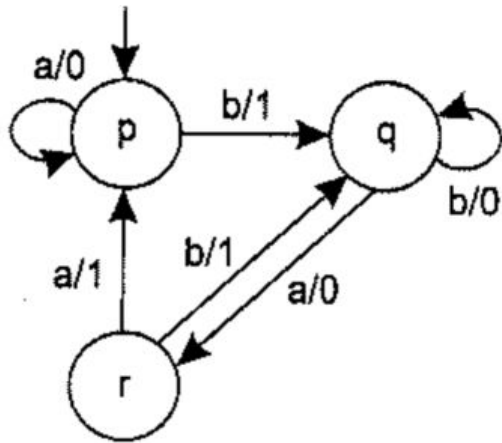
≡



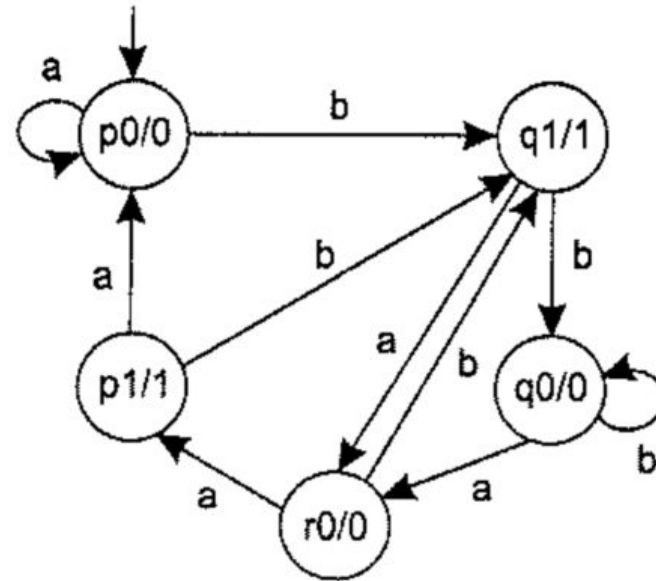
Автомат Мили

Из автомата Мура можно  
получить эквивалентный  
автомат Мили

# Автоматы Мили и Мура



≡



Автомат Мили

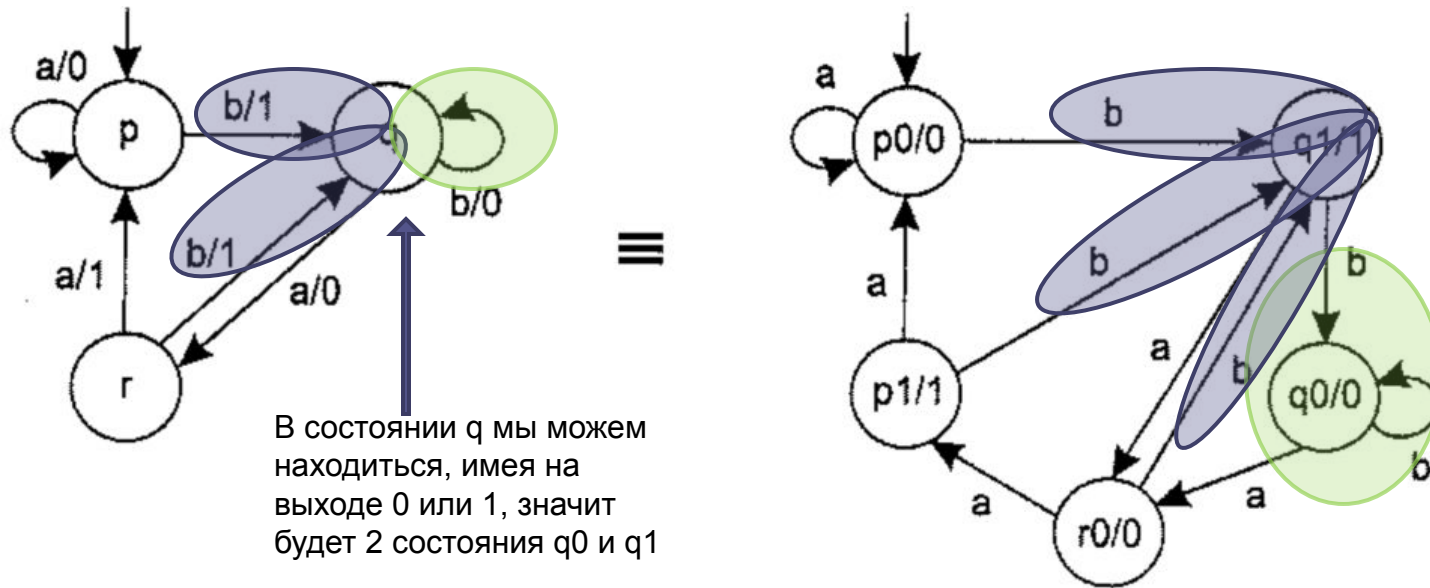


Автомат Мура

Из автомата Мили можно  
получить эквивалентный  
автомат Мура

Каждое состояние  $s$  автомата Мили расщепляется на несколько эквивалентных состояний, с каждым из которых связан один из выхдных символов

# Автоматы Мили и Мура



Автомат Мили



Автомат Мура

Из автомата Мили можно получить эквивалентный автомат Мура

Каждое состояние s автомата Мили расщепляется на несколько эквивалентных состояний, с каждым из которых связан один из выхдных символов

# Условия «автоматности» оператора

$\varphi_A: P \rightarrow Q$  - алфавитный оператор

1  $\varphi_A: P \rightarrow Q$  осуществляет однозначное отображение из  $P$  в  $Q$ .

2 Если  $P$  содержит слово  $p$ , то  $P$  содержит и все начальные отрезки слова  $p$ , включая пустое слово.

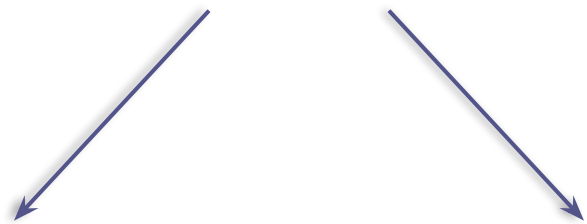
3  $\varphi_A: P \rightarrow Q$  сохраняет длину слова.

$$4 \begin{cases} p^k = p^{k-1}z(k) \\ q^k = q^{k-1}w(k) \\ \varphi_A(p^k) = q^k \end{cases} \Rightarrow \varphi_A(p^{k-1}) = q^{k-1}$$

Такой оператор называется *автоматным оператором*.

# Теория автоматов

Теория автоматов  
Состав теории



**Абстрактная теория**  
Математический аппарат теории автоматов, представляет связь с алгеброй и логикой.

**Структурная теория**  
Описывает способы реализации автомата при помощи заданного набора элементов.

Автоматы, рассматриваемые безотносительно их структуры, принято *абстрактными автоматами*.

Абстрактный автомат задаётся своим входным алфавитом, выходным алфавитом, множеством состояний и автоматным оператором.

# Теория автоматов и формальных языков

## Приложения теории автоматов

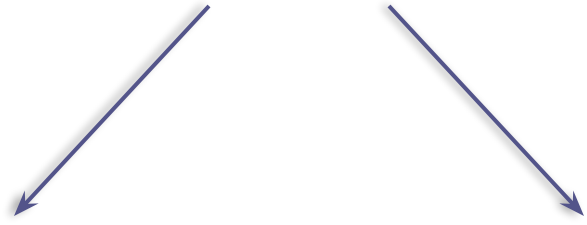
Институт Информационных  
Технологий  
ЧелГУ, 2010

A decorative graphic element consisting of several horizontal lines of varying lengths and colors (teal, white, and light blue) extending from the right side of the page towards the center.



# Классификация автоматов

**Автоматы**  
Классификация по основным функциям



**Автоматы – распознаватели**  
Отвечают на вопрос, принадлежит ли заданная последовательность символов какому-либо множеству.

**Автоматы – преобразователи**  
Преобразуют одну последовательность символов в другую последовательность символов.

# Распознаватель правильного идентификатора

Правильным идентификатором называется последовательность букв, цифр и символа подчёркивания, начинающаяся с буквы или символа подчёркивания.

`_a123`  
`Var75`  
`my_value` } Правильные идентификаторы

Пусть реализованы функции:

```
int isLetter(char ch);  
int isDigit(char ch);  
int isSmall(char ch);
```

$a(t - 1)$	digit	letter	small
$a_0$	-	$a_1$	$a_1$
$a_1$	$a_1$	$a_1$	$a_1$

# Распознаватель правильного идентификатора

Правильным идентификатором называется последовательность букв, цифр и символа подчёркивания, начинающаяся с буквы или символа подчёркивания.

Пусть реализованы функции:

```
int isLetter(char ch);  
int isDigit(char ch);  
int isSmall(char ch);
```

$a(t-1)$	digit	letter	small
$a_0$	-	$a_1$	$a_1$
$a_1$	$a_1$	$a_1$	$a_1$

Достаточно функций:

```
int isLetterOrSmall(char ch);  
int isDigit(char ch);
```

$a(t-1)$	digit	letter or small
$a_0$	-	$a_1$
$a_1$	$a_1$	$a_1$

# Распознаватель перечисления

$\{123, 65, 767, -43\}$   
 $\{+73, -2, 11, \}$

Допустимые входные последовательности

$a(t-1)$	{	}	sign	digit	,
$a_0$	$a_1$	-	-	-	-
$a_1$	-	┌	$a_2$	$a_2$	-
$a_2$	-	┌	-	$a_2$	$a_3$
$a_3$	-	┌	$a_2$	$a_2$	-

Можно ли представить данный автомат в виде комбинации каких-либо других двух автоматов?