

Алгоритмы

Маркова

Алгоритмы

и алгоритмические языки

goo.gl/c8puyx

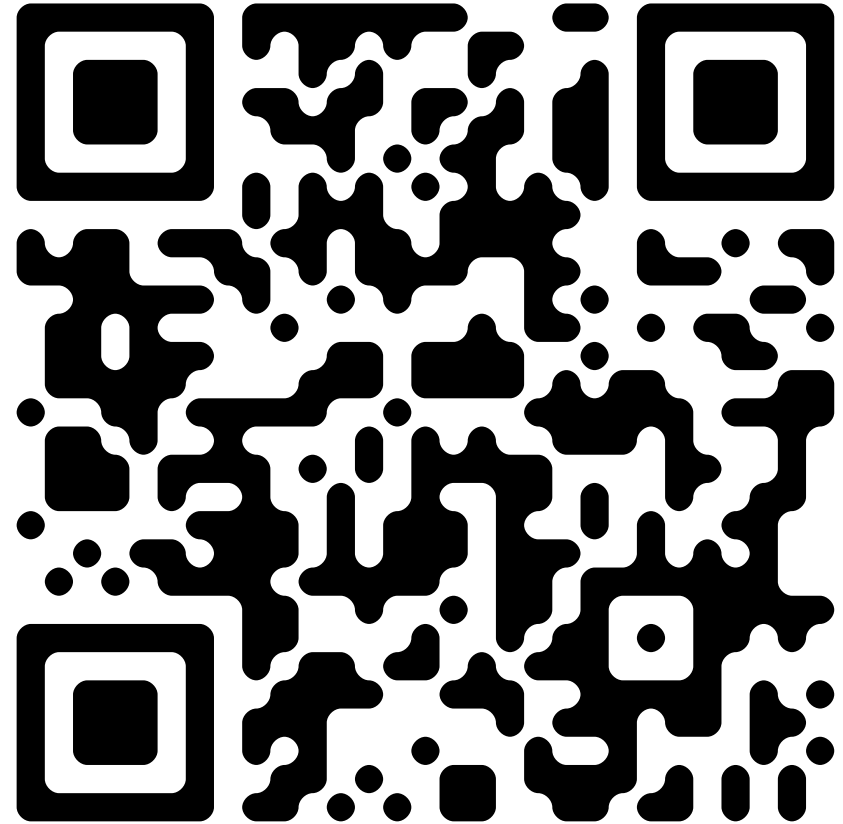
Лекция 3, 21 сентября, 2018

Лектор:

Дмитрий Северов, кафедра информатики 608 КПМ

dseverov@mail.mipt.ru

http://cs.mipt.ru/wp/?page_id=6077





Нормальные алгоритмы Маркова

1946 г.

Обозначения

Пусть $\sigma, \sigma', \alpha, \beta, \beta', \gamma$ - некоторые, возможно пустые, строки в алфавите V

причем $\sigma = \alpha\beta\gamma$ и $\sigma' = \alpha\beta'\gamma$

Тогда σ' есть результат подстановки β' в строку σ на место строки β .

Такая подстановка может быть задана функцией $f(\sigma, \alpha, \beta, \beta')$

Пример: $\sigma = \text{ааааа}, \beta = \text{аа}, \beta' = \text{b},$

Нормальная подстановка

Функция $f(\sigma, \beta, \beta')$ задаёт нормальную подстановку, заменяя самое левое вхождение строки β в строку σ на строку β' .

Пусть $V = \{a, b, \dots\}$ – основной алфавит,
 $V' = \{A, B, \dots\}$ – вспомогательный, причем $V \cap V' = \emptyset$
и $\beta, \beta' \in (V \cup V')^*$

Тогда $\beta \rightarrow \beta'$ – это простое правило,
а $\beta \rightarrow^1 \beta'$ – конечное правило подстановки.

Нормальный алгоритм Маркова

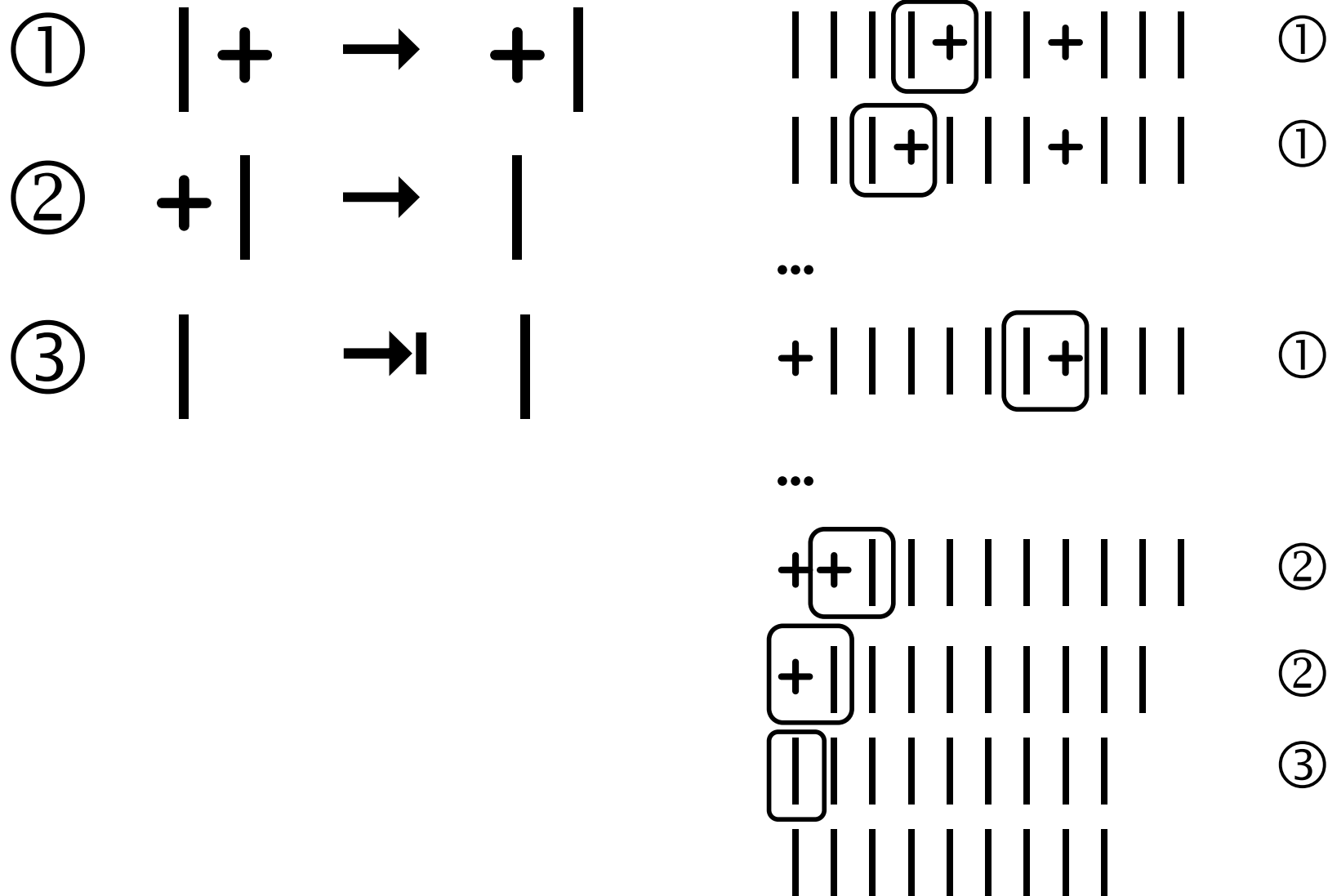
Упорядоченное множество подстановок видов

$$\beta \rightarrow \beta' \text{ и } \beta \rightarrow_i \beta'$$

Исполнитель, просматривая слово σ слева направо, пытается последовательно применить к нему правила из множества подстановок.

1. Если удастся применить правило вида $\beta \rightarrow \beta'$, то процесс повторяется со словом σ' ;
2. Если – вида $\beta \rightarrow_i \beta'$ – происходит результативный останов.
3. Если не удастся применить ни одно из правил – останов не результативный (аварийный).

Пример 1: сложение унарных чисел



Пример 2

Удалить из непустого слова в алфавите {a,b} первый символ. Пустое слово не менять.

a → |

bbb|abab → bbbbab

b → |

→ *

*a → |

bbbabab → *|bbbabab → **bbbabab...

*b → |

Пример 2

Удалить из непустого слова в алфавите {a,b} первый символ. Пустое слово не менять.

→ *

*a →| bbbabab → *bbbabab → **bbbabab...

*b →|

*a →|

*b →| bbbabab → *bbbabab → bbabab

→ *

Пример 2

Удалить из непустого слова в алфавите {a,b} первый символ. Пустое слово не менять.

*a → |

*b → | → * → ** ...

→ *

*a → |

*b → | bbbabab → *bbbabab → bbabab

* → | → * →

→ *

Композиция алгоритмов Маркова

- Пусть заданы два алгоритма R и S .
- Задача: построить алгоритм, который будет выполнять $S(R(\sigma))$.
- Проблема: как заблокировать подстановки R после того, как завершён R и заработал S

Композиция алгоритмов Маркова

1. $\xi\alpha \rightarrow \alpha\xi$ n правил
2. $\alpha\xi \rightarrow \alpha\xi$ n правил
3. $\xi\eta \rightarrow \xi\eta$ n^2 правил
4. $\xi\beta \rightarrow \beta\xi$ n правил
5. $\beta\xi \rightarrow \beta\xi$ n правил
6. $\xi\eta \rightarrow \xi\eta$ n^2 правил
7. $\alpha\beta \rightarrow \epsilon$
8. S_{β}
9. R_{α}

Композиция алгоритмов Маркова

1. Удвоим алфавит, добавив для каждого символа алфавита ξ его близнеца $\underline{\xi}$
2. Добавим еще два символа α и β , которые не входили в исходный алфавит.
3. Преобразуем R в R_α заменой правил вида $\dots \rightarrow^1 \zeta$ на $\dots \rightarrow \alpha\zeta$
4. Преобразуем S в S_β заменой
 1. ξ на $\underline{\xi}$
 2. правил вида $\dots \rightarrow^1 \zeta$ на $\dots \rightarrow \zeta\beta$

Пример композиции

R: перевод
двоичного
в унарное

① | 0 → 0 | |
 ② 1 → 0 |
 ③ 0 →
 → I

S: суммирование

④ + →
 ⑤ → I

1 0 1 + 1 1	②
0 0 1 + 1 1	②
0 0 1 + 1 1	②
0 0 0 + 1 1	①
0 0 0 + 1 1	①
0 0 0 + 1 1	②
0 0 0 + 0 1	②
0 0 0 + 0 0	①
0 0 0 + 0 0	③
... +	④
	⑤

Формальная схема

$ \alpha \rightarrow \alpha $	$\underline{\pm} \rightarrow$
$+\alpha \rightarrow \alpha+$	$\alpha \rightarrow \alpha\beta$
$\alpha \rightarrow \alpha\underline{ }$	
$\alpha+ \rightarrow \alpha\underline{+}$	
$\underline{ }+ \rightarrow \underline{ }\underline{+}$	$ 0 \rightarrow 0 $
$\underline{ } \rightarrow \underline{ }\underline{ }$	$1 \rightarrow 0 $
$\underline{+} \rightarrow \underline{+}\underline{ }$	$0 \rightarrow$
$\underline{+}+ \rightarrow \underline{+}\underline{+}$	$\lambda \rightarrow \alpha$
$\underline{ }\beta \rightarrow \beta\underline{ }$	
$\beta\underline{ } \rightarrow \beta $	
$ \underline{ } \rightarrow $	
$\alpha\beta \rightarrow \alpha$	

11+1

0|1+1

0|0|+1

00||+1

00||+0|

0||+0|

||+0|

||+|

$\alpha| | | + |$

$\alpha\underline{|} | | + |$

$\alpha\underline{|}\underline{|} | + |$

$\alpha\underline{|}\underline{|}\underline{|} + |$

$\alpha\underline{|}\underline{|}\underline{|} + |$

$\alpha\underline{|}\underline{|}\underline{|} + |$

$\alpha\underline{|}\underline{|}\underline{|}$

$\alpha\beta\underline{|}\underline{|}\underline{|}$

$\alpha\beta| \underline{|}\underline{|}$

$\alpha\beta| | \underline{|}$

$\alpha\beta| | | \underline{|}$

$\alpha\beta| | |$

$| | |$

Вместо
 $\underline{\lambda} \rightarrow \beta$

Алгоритм Маркова сложения двоичных чисел

- ① $\beta \underline{1} \rightarrow 1\beta$
- ② $\beta \underline{0} \rightarrow 0\beta$
- ③ $\beta \rightarrow \uparrow$

Перевод в унарную

$$|0 \rightarrow 0|| \quad \text{①}$$

$$1 \rightarrow 0| \quad \text{②}$$

$$0 \rightarrow \quad \text{③}$$

Сложение

$$+ \rightarrow \quad \text{④}$$

$$| \rightarrow \perp \quad \text{⊗}$$

Перевод в двоичную

$$\alpha \underline{\perp} \rightarrow \perp \alpha \quad \text{⑤}$$

$$\alpha \perp \rightarrow \underline{1} \quad \text{⑥}$$

$$\alpha \rightarrow \underline{0} \quad \text{⑦}$$

$$\perp \rightarrow \alpha \perp \quad \text{⑧}$$

$$\rightarrow \beta \quad \text{⑨}$$

101+111 ②

Пример

0 | 01+111 ①

00 | | 1+111 ②

00 | | 0 | +111 ①

00 | 0 | | | +111 ①

000 | | | | +111 ②

000 | | | | +0 | 11 ②

000 | | | | +0 | 0 | 1 ①

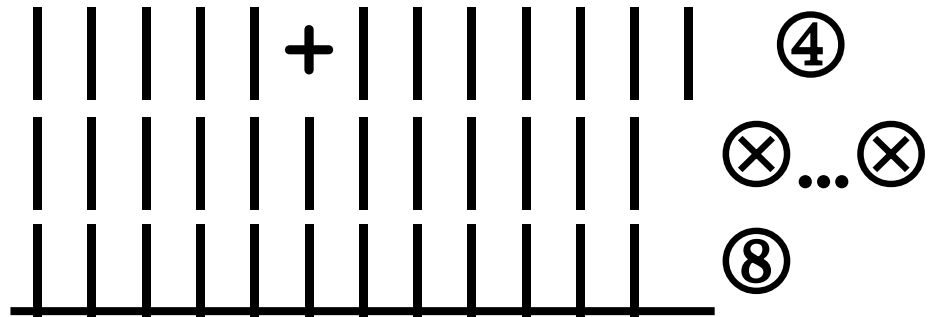
000 | | | | +00 | | | 1 ②

000 | | | | +00 | | | 0 | ②

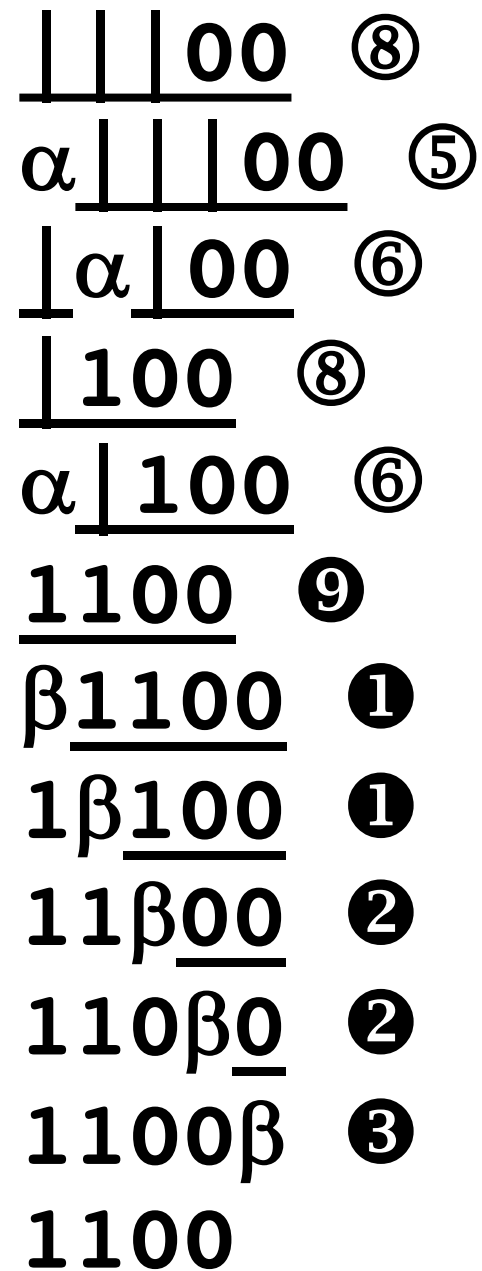
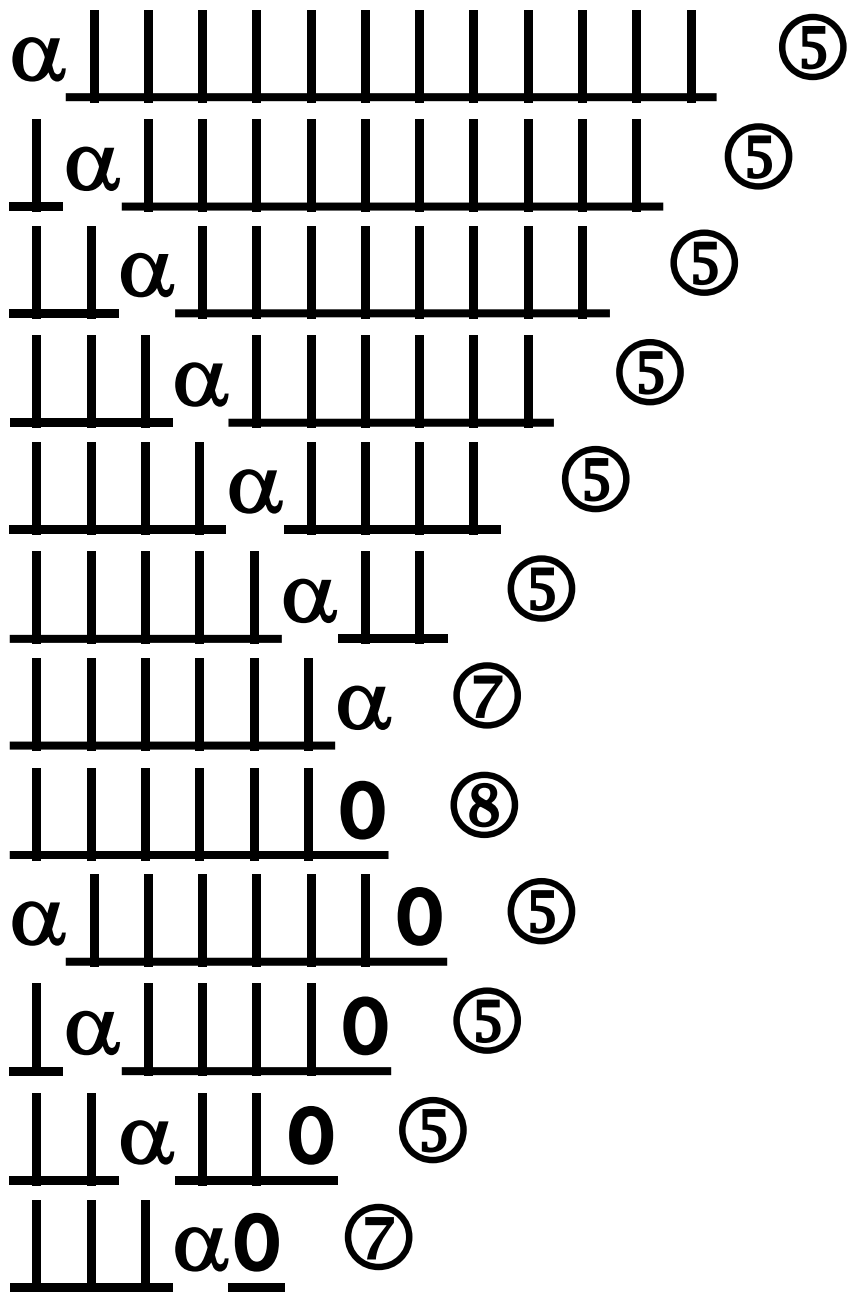
000 | | | | +00 | | 0 | | | ②

000 | | | | +00 | 0 | | | | ②

000 | | | | + 000 | | | | | | | ③...③



Пример



Проблема самоприменимости

- › Невозможно построить НАМ, который для любого другого НАМ выносил бы решение о том, произойдет или нет останов этого НАМ при его работе над данными, представляющими описание этого НАМ.

Проблема самоприменимости

Самоприменимый

$$\textcircled{1} a \rightarrow b$$

поскольку

$$a \rightarrow b \textcircled{1}$$

$$b \rightarrow b$$

Несамоприменимый

$$\textcircled{1} a \rightarrow ab$$

поскольку

$$a \rightarrow ab \textcircled{1}$$

$$ab \rightarrow ab \textcircled{1}$$

$$abb \rightarrow ab \textcircled{1}$$

...