

# Домашнее задание №1

## Логика и Теория Алгоритмов

Козырнов Александр Дмитриевич  
ИУ7-42Б  
Вариант 6

11 марта 2024 г.

### Содержание

<b>1</b>	<b>Условие задачи</b>	<b>2</b>
<b>2</b>	<b>Решение задачи</b>	<b>2</b>
<b>3</b>	<b>Прогонка</b>	<b>3</b>
3.1	Положительный результат работы НА . . . . .	3
3.2	Отрицательный результат работы НА . . . . .	3

# 1 Условие задачи

Построить НА, который аннулирует все слова вида  $x\$x$ , где  $x \in \{a, b\}^*$ , а  $\$ \notin \{a, b\}$

# 2 Решение задачи

Пусть слово задано как внешний параметр:

$$u = u(1)u(2) \dots u(k), \quad k \geq 1$$

Назовем наш НА *DoubleDel*. Его схема:

$$DoubleDel : \begin{cases} \nabla \xi \rightarrow \bullet \xi // \xi \in \{a, b\} & (1) \\ \nabla \rightarrow \gamma // \nabla, \gamma \notin \{a, b\} & (2) \\ \xi \gamma \rightarrow \gamma & (3) \\ \$ \gamma \rightarrow \gamma & (4) \\ \gamma \rightarrow \bullet & (5) \\ \# \$ u \rightarrow \$ u(1)u(2) \dots u(k) \nabla & (6) \\ \# \xi \rightarrow \bullet \xi & (7) \\ \# \$ \rightarrow \bullet \$ & (8) \\ \square u \rightarrow u(1)u(2) \dots u(k) \# & (9) \\ \square \xi \rightarrow \bullet \xi & (10) \\ \square \rightarrow \bullet & (11) \\ \rightarrow \square & (12) \end{cases}$$

Следует уточнить, какой символ что делает.

Параметр  $\xi$  обозначает буквы нашего алфавита  $\{a, b\}$ . Символ  $\square$  ищет нахождение первого вхождения слова  $u$ . В начале программы символ  $\square$  появляется в начале всего слова. У нас такие варианты действий: у нас пустое слово; мы нашли букву; мы нашли слово  $u$ ; мы встретили разделитель  $\$$ . В первом случае сработает формула (11). Во втором случае сработает формула (10). В Третьем случае нахождения слова сработает формула (9), причем она имеет приоритет над формулой (10). В четвертом случае сработает та же формула (11).

Если мы выполнили формулу (9), то мы начинаем работать со знаком  $\#$ . Замечу, что добавляем его в конец слова  $u$ , так как нет смысла переносить его вручную побуквенно. В этом состоянии, когда  $\#$  в конце первого слова  $u$ , у нас есть такие варианты действий: встретили букву алфавита, встретили разделитель  $\$$  без второго слова или встретили разделитель с вхождением второго слова. Если встретили букву алфавита в конце слова  $u$ , то завершается работа НА формулой (7). Во втором случае встретим формулу (8), причем разделитель мы убирать не станем. В успешном варианте встречаем формулу (6) и ставим знак  $\nabla$  в конец второго слова.

Если мы успешно нашли второе вхождение слова  $u$ , то мы начинаем работать с символом  $\nabla$ . У нас снова несколько вариантов действий: мы находимся в конце слова (следующий символ - пробел) или встретили букву после второго вхождения слова. В первом случае мы вводим 'ластик' -  $\gamma$ , в ином случае завершаем работу алгоритма с удалением  $\nabla$ . Замечу, что  $\nabla$  можно спокойно заменить на  $\#\#$ .

Формулы (3) и (4) стирают все буквы, в том числе и разделитель. Формула (5) стирает сам ластик и заканчивает работу НА.

При подаче пустого слова сработает формула (12)  $\rightarrow$  (11). При подаче слова без вхождения слова  $u$  алгоритм завершает работу на (12)  $\rightarrow$  (10) формулах. В случае вхождения какого-то слова  $\omega$  до слова  $u$  завершаем НА формулой (12)  $\rightarrow$  (10).

### 3 Прогонка

#### 3.1 Положительный результат работы НА

В случае успеха слово должно быть аннулировано.

##### Пример 1.

Пусть  $u = a$

Тогда

$$\begin{aligned} DoubleDel(a\$a) = \\ = a\$a \vdash \square a\$a \vdash a\#\$a \vdash a\$a \nabla \vdash a\$a \gamma \models^3 \gamma \vdash \cdot \lambda \end{aligned}$$

(12) (9) (6) (2) (3),(4) (5)

##### Пример 2.

Пусть  $u = abba$

Тогда

$$\begin{aligned} DoubleDel(abba\$abba) = \\ = abba\$abba \vdash \square abba\$abba \vdash abba\#\$abba \vdash \\ \vdash abba\$abba \nabla \vdash abba\$abba \gamma \models^9 \gamma \vdash \cdot \lambda \end{aligned}$$

(6) (2) (3),(4) (5)

#### 3.2 Отрицательный результат работы НА

В случае неуспеха нормальный алгоритм вычислит тождественную функцию

##### Пример 1.

Пусть  $u = \{a, b\}^*$

Подадим на вход НА пустое слово. Получаем

$$DoubleDel(\lambda) = \lambda \vdash \square \vdash \cdot \lambda$$

(12) (11)

##### Пример 2.

Пусть  $u = \{a, b\}^*$ .

Подадим на вход НА слово вида  $\lambda\$ \omega$ , где  $\omega \in \{a, b\}^*$ . Не исключено, что  $\omega = u$ . Получаем

$$DoubleDel(\$abcde) = \$abcde \vdash \square \$abcde \vdash \cdot \$abcde$$

(12) (11)

Как можем заметить, НА  $DoubleDel$  реагирует на такую ситуацию аналогично пустому слову.

### Пример 3.

Пусть  $u = abba$ .

Подадим на вход НА такое до разделителя  $\$$  слово  $\omega$ , чтобы в него входило слово  $u$ , причем справа. То есть в виду имеется такое:  $\omega = \delta u$ , где  $\delta \in \{a, b\}^+$ , причем  $\delta \neq u$ . Получаем

$$DoubleDel(aabba\$abba) = aabba\$abba \vdash \square_{(12)} aabba\$abba \vdash \cdot_{(10)} aabba\$abba$$

Это работает потому, что знак  $\square$  неподвижен и находится всегда в начале слова. Он проверяет сначала вхождение слова  $u$ , но если его не находит, то стирается и НА завершает свою работу.

### Пример 4.

Пусть  $u = ab$ .

Подадим на вход НА такое до разделителя  $\$$  слово  $\omega$ , чтобы в него входило слово  $u$ , причем слева. То есть в виду имеется такое:  $\omega = u\delta$ , где  $\delta \in \{a, b\}^+$ , причем не исключено  $\delta = u$ . Получаем

$$DoubleDel(abab\$ab) = abab\$ab \vdash \square_{(12)} abab\$ab \vdash ab\#_{(9)} ab\$ab \vdash \cdot_{(7)} abab\$ab$$

Алгоритм нашел вхождение первого слова. Однако он встретил посторонние буквы после вхождения  $u$ . Неважно, будет ли это вторым вхождением слова, так как для этого у нас есть разделитель. В этом случае сработала формула (7)  $\# \xi \rightarrow \cdot \xi$ , которая просто стёрла знак решетки и завершила работу НА.

### Пример 5.

Пусть  $u = ab$ .

Подадим на вход НА такое слово с разделителем, чтобы он прочитал первое вхождение, но не смог прочесть второе.

$$DoubleDel(ab\$aab) = ab\$aab \vdash \square_{(12)} ab\$aab \vdash ab\#_{(9)} \$aab \vdash \cdot_{(8)} ab\$aab$$

НА нашел вхождение первого слова и создал конструкцию вида  $\# \$$ , но не нашел вхождения второго слова сразу после нее. Вместо этого он нашел слово вида  $\omega \in \{a, b\}^*$ , причем  $\omega$  не начинается со слова  $u$ . Результат работы будет таким же, если  $\omega = \lambda$ , то есть слово вида  $u\$ \lambda$ .

### Пример 6.

Пусть  $u = ab$ .

Подадим на вход НА такое слово с разделителем, чтобы он прочитал первое и второе вхождение слов  $u$ , но нашел 'хвост' после второго вхождения.

$$DoubleDel(ab\$aba) = ab\$aba \vdash \square_{(12)} ab\$aba \vdash ab\#_{(9)} \$aba \vdash ab\$ab \nabla_{(6)} a \vdash \cdot_{(1)} ab\$aba$$

По аналогии с  $\square$  знак  $\nabla$  тоже неподвижен и всего лишь проверяет конечный символ слова. В нашем случае он встретил не пробельный символ, отчего по формуле (1) завершается работа НА и стирается треугольник.

**Пример 7.**

Пусть  $u = ab$

Подадим на вход НА слово без разделителя вовсе, но при этом на входе будет двойное слово  $u$ . Получаем

$$DoubleDel(abab) = abab \vdash_{(12)} \square abab \vdash_{(9)} ab \# ab \vdash_{(7)} \bullet abab$$