

# Формальные языки. Домашнее задание №4

Горбунов Егор Алексеевич

6 апреля 2016 г.

**Задание №1** Преобразовать в нормальную форму Хомского однозначную грамматику арифметических выражений, найти реализацию алгоритма СУК, запустить ее на полученной грамматике, трех различных корректных входах и трех некорректных, визуализировать полученные деревья вывода и записать сообщения об ошибках (прислать результаты работы парсера в pdf).

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T * P \mid P$$

$$P \rightarrow (E) \mid n$$

**Решение:** Грамматика у нас такая:

$$\Sigma = \{+, *, (, ), n\}$$

$$N = \{E, T, P\}$$

$$S = E$$

$\epsilon$ -продукций в грамматике нет, а также нетрудно видеть, что абсолютно все символы грамматики достижимые и порождающие.

Найдём все цепные пары индуктивно:

$$\begin{array}{ccccc} & & & & (E, E) \\ & & & & (E, T) \\ (E, E) & & (E, E) & & (E, P) \\ (T, T) \Rightarrow & (E, T) & \Rightarrow & (T, T) & \Rightarrow (T, T) \\ (P, P) & (T, P) & & (P, P) & (P, P) \end{array}$$

Теперь удалим цепные пары и добавим нужные productions в грамматику:

$$E \rightarrow E + T \mid T * P \mid (E) \mid n$$

$$T \rightarrow T * P \mid (E) \mid n$$

$$P \rightarrow (E) \mid n$$

Теперь приведём к нормальной форме Хомского. Для начала избавимся от терминалов в правых частях productions, длины больше 1:

$$L \rightarrow ($$

$$R \rightarrow )$$

$$X \rightarrow *$$

$$O \rightarrow +$$

$$E \rightarrow E O T \mid T X P \mid L E R \mid n$$

$$T \rightarrow T X P \mid L E R \mid n$$

$$P \rightarrow L E R \mid n$$

И разобьём наконец слишком длинные правила:

$$L \rightarrow ($$

$$R \rightarrow )$$

$$X \rightarrow *$$

$$O \rightarrow +$$

$$E \rightarrow E D \mid T M \mid L C \mid n$$

$$T \rightarrow T M \mid L C \mid n$$

$$P \rightarrow L C \mid n$$

$$D \rightarrow O T$$

$$M \rightarrow X P$$

$$C \rightarrow E R$$

Ниже изображены деревья (в силу особенностей JFLAP было добавлено правило  $S \rightarrow E$  и круглые скобки  $()$  заменены на квадратные  $[]$ ).

Все парсеры, которые я пробовал, не выдают информации об ошибке при неправильных вводах, поэтому картинки с «недостроенными» деревьями не прикреплены. Но ясно, что это были бы деревья, без верхушки :)

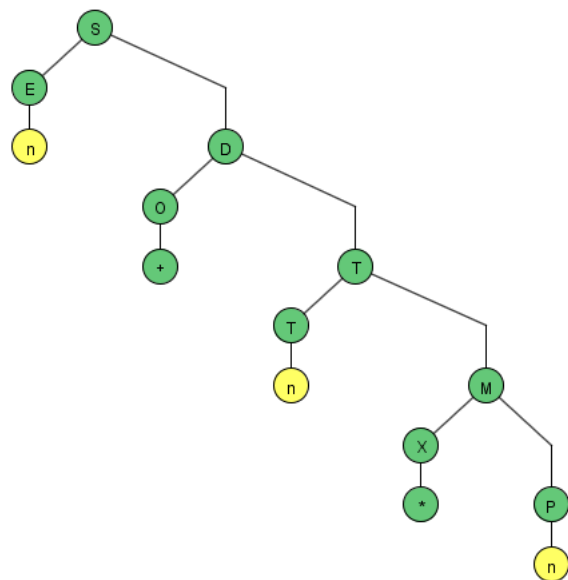


Рис. 1: Дерево выражения  $n + n * n$

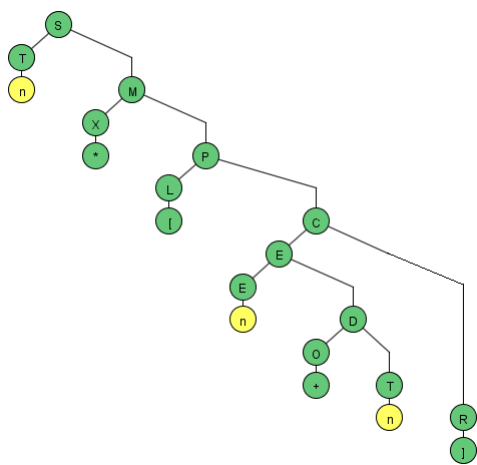


Рис. 2: Дерево выражения  $n * [n + n]$

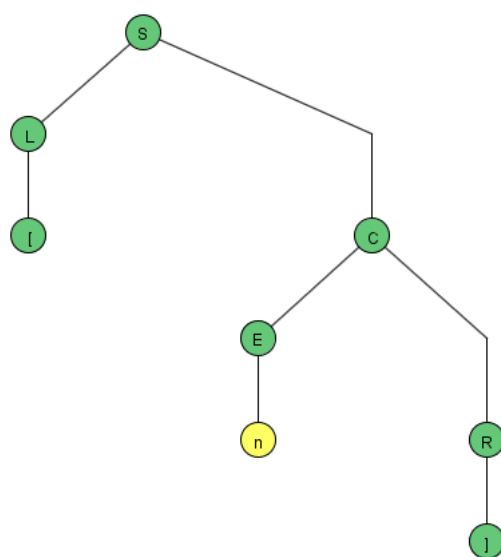


Рис. 3: Дерево выражения  $[n]$