Алексей Косенко

ВШЭ ПМИ, 2022 г.

1 Формальные языки, практика 1.

1.1 Привести грамматику для языка палиндромов.

Язык палиндромов - $\{ww^r|w \in \{0,1\}*\}$. Это язык, в котором цепочки начинаются и заканчиваются одинаково.

$$G = \{\{0,1\}, \{S\}, P, S\}$$

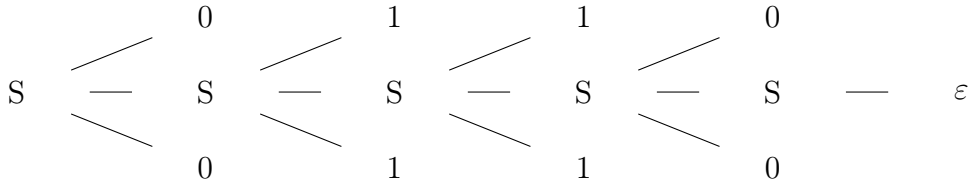
Для цепочек четной длины в начало и в конец будем ставить одинаковые терминальные значения, то есть открывающую и закрывающую скобку сразу.

$$P : S \to 0S0 \mid 1S1 \mid \varepsilon$$

Если нам нужна ещё цепочка нечётной длины, то добавим элемент из алфавит с пустым символом, иначе просто ε.

$$P : S \to 0S0 \mid 1S1 \mid 1\varepsilon \mid 0\varepsilon \mid \varepsilon$$

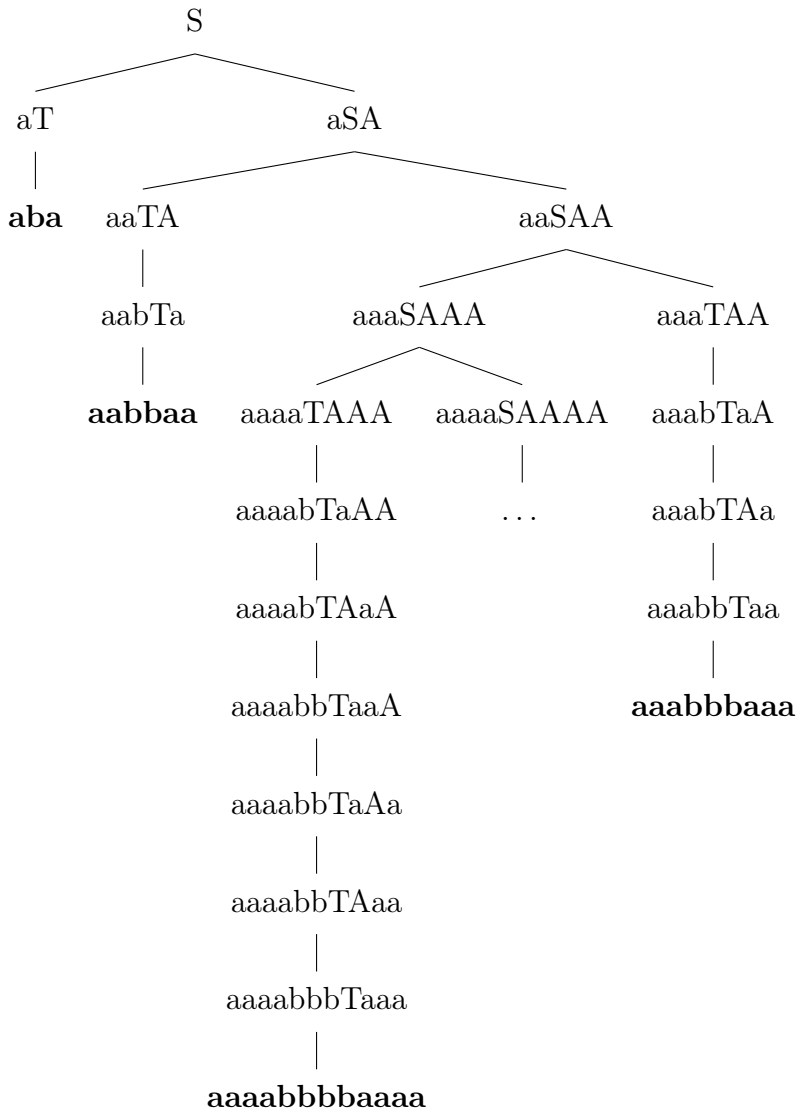
Приведем дерево вывода строки 01100110 в нашей грамматике, длина которой равна 8.



$$S \to 0S0 \to 01S10 \to 011S110 \to 0110S0110 \to 0110\varepsilon0110 \to 01100110$$

1.2 Описать язык, порождаемый грамматикой.

$$P : S \to aSA \mid aT$$
$$TA \to bTa$$
$$aA \to Aa$$
$$T \to ba$$



Итак, правила предложенной грамматики описывают цепочки формата: $a^n b^n a^n$, где $n \in \mathbb{N}$.

1.3 Отрывок из спецификации Python.

Вот ссылка на документ.

- В PEP 572 был введён оператор моржа (*walrus operator*) - попытка разграничить оператор присваивания и тоже самое, но в контексте большого выражения. Он не только снижает сложность кода, но и улучшает его чтение.

$$a = (1 + (n := 10))$$
$$print(a, n)$$

- Типизация динамически типизированного языка была введена в PEP 483 с помощью стандартной библиотеки *typing*, другими словами - type hinting.

Чем-то похоже на TypeScript, но без компилятора, а только с *mypy*. Есть даже доклады, где сравнивается а-ля типизация между JS и Python. Такой "хороший костыль"сделал код быстро читаемым из-за сохранения контекста типов и объектов.

a: int = 10

T = TypeVar("T") # дженерики

def bar(a: T) → T: ...

def foo(a: List[int, ...]) → Union[None, Tuple[People]]: ...

- Распаковщик списков (*последовательностей*) и словарей. Локаничность и точка.

  a, *b, c = range(10)

  def baz(..., *arg, **kwargs) → ... :

  [*array]; $\{**dict1, **dict2\}$; print(*array)

- F-строки, ellipsis, mock-функции (*pass*) и многое другое.