编译原理作业 (5)

姓名: _____   **学号:** _____

2024 年 05 月 05 日

请独立完成作业，不得抄袭。
若得到他人帮助，请致谢。
若参考了其它资料，请给出引用。
鼓励讨论，但需独立书写解题过程。
允许并鼓励使用 ChatGPT 等工具，但需明确说明使用方式。

Semantics of Context-Free Languages

by

DONALD E. KNUTH

California Institute of Technology

ABSTRACT

"Meaning" may be assigned to a string in a context-free language by defining "attributes" of the symbols in a derivation tree for that string. The attributes can be defined by functions associated with each production in the grammar. This paper examines the implications of this process when some of the attributes are "synthesized", i.e., defined solely in terms of attributes of the *descendants* of the corresponding nonterminal symbol, while other attributes are "inherited", i.e., defined in terms of attributes of the *ancestors* of the nonterminal symbol. An algorithm is given which detects when such semantic rules could possibly lead to circular definition of some attributes. An example is given of a simple programming language defined with both inherited and synthesized attributes, and the method of definition is compared to other techniques for formal specification of semantics which have appeared in the literature.

1 作业 (必做部分)

题目 1

考虑如下文法 G,

$$S \to L.L \mid L$$
$$L \to LB \mid B$$
$$B \to 0 \mid 1$$

请设计语法制导的翻译方案, 计算 S 产生的二进制小数对应的十进制值。如，$101.101_2 = 5.625_{10}$。你需要自行定义合适的属性。

解答:

题目 2

考虑如下文法 G,

$$S \rightarrow (L) \mid a$$
$$L \rightarrow L, S \mid S$$

请设计语法制导的翻译方案, 完成下列任务。你需要自行定义合适的属性。

(1) 计算每个 a 的嵌套深度。例如, 在句子 $(a, (a, a))$ 中, 每个 a 的嵌套深度分别为 $1, 2, 2$。

(2) 计算每个 a 的位置。例如, 在句子 $(a, (a, (a, a), (a)))$ 中, 每个 a 的位置分别为 $2, 5, 8, 10, 14$。

解答:

题目 3

考虑文法 $\mathcal{G}_D$ (描述了 C 语言的声明语法):

$$D \rightarrow D\ (\ ) \tag{1}$$
$$D \rightarrow D\ [\ ] \tag{2}$$
$$D \rightarrow *\ D \tag{3}$$
$$D \rightarrow (D) \tag{4}$$
$$D \rightarrow \mathbf{id} \tag{5}$$

　　请设计语法制导的翻译方案, 为 **id** 的类型提供英文/中文说明文档 (由于中英文语法结构的不同, 输出英文说明文档可能更容易; 例子见下页)。

　　你需要自行定义合适的属性, 请**指明**哪些是综合属性, 哪些是继承属性。

　　注意: 对于处于最外层的 $D[]$, 不需要输出 array of, 只需输出 array。
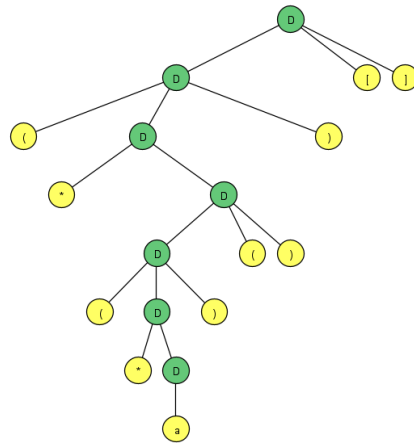
解答:

图 1: `(*(*a)())[]`: declare `a` as pointer to function returning pointer to array
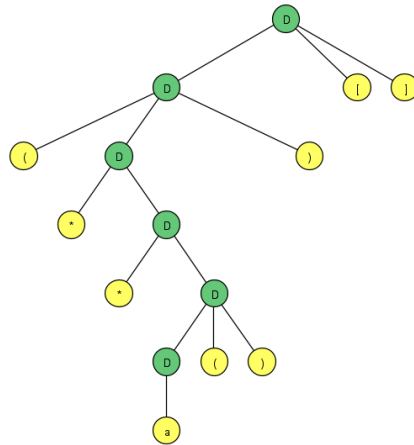

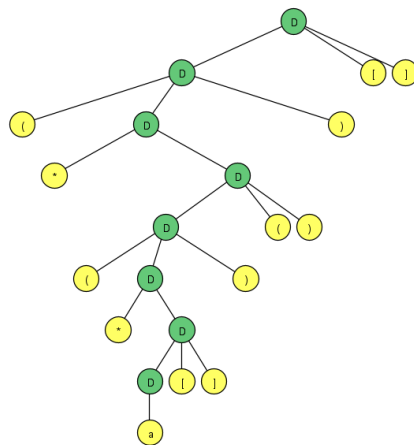图 2: `(**a())[]`: declare a as function returning pointer to pointer to array

图 3: `(*(*a[])())[]`: declare `a` as array of pointer to function returning pointer to array