ISSN 0005-1055, Automatic Documentation and Mathematical Linguistics, 2008, Vol. 42, No. 4, pp. 212–220. © Allerton Press, Inc., 2008.
Original Russian Text © V.S. Vykhovanets, 2008, published in Nauchno-Tekhnicheskaya Informatsiya, Seriya 2, 2008, No. 7, pp. 6–14.

Description of the Semantics of Context-Free Languages by the Mathematical Induction Method

V. S. Vykhovanets

Received May 05, 2008

Abstract—Context-free languages are studied. The semantics of a context-free language is determined by a sequential description of the input rules of formal grammar and inductive expression of each rule through basic and previously determined semantic categories presented as texts on the language under consideration. An external interpreter is used to determine basic semantic categories. A posting grammatical analysis aimed at increasing the efficiency of text processing is discussed.

DOI: 10.3103/S000510550804002X

INTRODUCTION

Presently there is no universal approach for the formal description of semantics. Numerous models and methods have been developed, such as:

* grammatical models based on the addition of extensions to a language-determining grammar (attribute grammars [1], imperative, or operational method [2], Vienna method [3], W-grammars [4]);

* applicative models wherein a function definition is directly constructed that is computed by each program written in this language (axiomatic methods [5], method of denotation semantics [6], method of functional semantics [7]);

* specification models whereby the relations between different language functions are described, and if a program creates these relations then it is correct relative to a specification (specification method [8], algebraic method [9].

In all these methods the semantics are described by distinguishing a meaning typology, that is, meaning categories (semantic categories and primitives) through which and by which a more complicated meaning is specified.

The three most widely used methods used to describe the semantics of context-free languages have been distinguished [10], viz., the production (deduction, axiomatic), denotation (mathematical), and operation methods. These are the only potentially applicable ones. The last fact can be attributed to their flexibility and the possibility of extending the set of semantic categories (mathematical objects, commands of a virtual machine, etc.) by which semantic description is executed.

The production method is based on predicate calculus where the computation result is described through the relationship of variables before and after application of language constructions. Production semantics is based on mathematical logic, is human-oriented and is designed for proving a program's correctness rather than for formal semantics definition. Hoare's axiomatic method [5] and Floyd's method of inductive statements [11] exemplify the production method for semantics description.

The denotation method is based on functional computations wherein inbuilt language operations are reflected as monosemantic mathematical objects, which are further used to describe the semantics of language constructions [12]. In particular, the given approach is illustrated by Scott's theory of computations based on semantic domains [13] where the first standard and definable (finite) domains are listed, and then, through the definition of constructors, the semantics of language constructions is specified as a formula over domains and constructors. Note that denotation type of language description is used for generation of compilers, however, no compiler has been created so far by this method [14].

In the operational method language operations are described through commands of an abstract machine, such as an SECD machine [15] or a categorical machine [16]. As practice shows, descriptions based on low-level simulation are useless, since they are very cumbersome and inconvenient for interpretation [16].

The operation-type description of semantics of context-free languages in the YACC system of generation of syntactical analyzers has found the widest application in practice [17, 18]. A context-free grammar is given to the program input, whose every production is related to an action described in the Си programming language and executed each time when the node of the parse tree is created corresponding to this production.

All the methods listed for semantics definition turned out to be inefficient for both users and language developers [8]. The operational method is fairly convenient to execute and useful for a developer but it is of insignificance value for a user since the descriptions generated by it comprise many unnecessary details. A