A tool for teaching LL and LR parsing algorithms

César García-Osorio
University of Burgos
Avda. Cantabria s/n
09006, Burgos, Spain
cgosorio@ubu.es

Carlos Gómez Palacios
University of Burgos
Avda. Cantabria s/n
09006, Burgos, Spain
carlosgomezpalacios@gmail.com

Nicolás García-Pedrajas
University of Córdoba
Campus Universitario de Rabanales
14071, Córdoba, Spain
npedrajas@uco.es

ABSTRACT

In this paper we present a tool to assist in teaching top-down and bottom-up analysis algorithms. The tool provides simulation for the following analysis algorithms: LL, SLR, LALR and LR. During the simulation the student can simultaneously see the pending input, the analysis stack and tables, the generated output and the parse tree.

Categories and Subject Descriptors
F.4.2 [Mathematical Logic and Formal Languages]: Grammars and other Rewriting Systems—grammar type, parsing

General Terms
Algorithms, Languages, Theory

Keywords
LL parsing, LR parsing, interactive teaching tool.

1. INTRODUCTION

Analysis algorithms are those used in compilers for analyzing the structure of the input program. A common problem when explaining analysis algorithms is that the examples that showed how they work are either to simple to show all subtleties of the algorithm or to long and tedious. Students often are more concerned about copying what the lecture writes on the blackboard than in understanding the process. Besides, is not uncommon that students made mistakes taking theirs notes and the examples turn into an obstacle instead of an assistance in understanding the algorithm. On the other hand, computer science students usually use the computer in other subjects. Here, we present BURGRAM, a tool for learning and teaching analysis algorithms.

Although there are other tools [1, 2, 3] similar to BURGRAM, BURGRAM solves some of their deficiencies: lack of interactivity and poor graphical user interface. With BURGRAM is possible to execute the algorithms step by step or undo these steps. BURGRAM shows simultaneously the parsing table, the stack contents, the pending input and the parse tree. The lecture or the student can choose in which elements to focus by making them to fill all the application window, while the rest remain hidden. Finally, BURGRAM is able to generate a report of the analysis process in different formats: PDF, RTF and HTML. The student can use them for subsequent reference adding them to the notes taken in classroom. This is a functionality that other tools do not have.

BURGRAM has been programmed using Java, the lexical and syntax analysis of the grammars is done with the tool ANTLR [5] and the automata and parse tree graphics with the library GRAPPA [3], and the report generation with the library iText [4].

2. CONCLUSION

We have presented a tool that allows the students the interaction with analysis algorithms. This is always better than just using pen and paper. The student can experiment with the algorithm and watch the evolution of the algorithm step by step. The lecturer can use the tool as support during the explanations of algorithms.

One of the most interesting characteristics of the tool is the report generation. The student can add this reports to his notes as a reference for later study.

The tool commented here is available for downloading in the following link http://pisuerga.inf.ubu.es/cgosorio/BURGRAM.

3. REFERENCES