ESnippets, a tool set for generating error snippets in the Merr format from YACC/Bison .y files

John Goettsche

May 15, 2014

Abstract

A tool set for generating Merr error snippets from a YACC/Bison .y file.

1 Introduction

One of the tedious tasks of an author of a programming language is the production of meaningful error messages for the developers who use the language. Anyone who has spent time working developing software is all too aware that they are capable of making their fair share syntactic errors. When a syntax error exists in some code, it is helpful to have a description of the error so that it can be easily identified and corrected.

The purpose of this directed study was to produce a software tool that reads in YACC/Bison .y files, construct a data structure representing the context free grammar, and use it to generate a set of example syntax error fragments in Merr format that exercise the entire grammar.

Before I could go through the process of implementing the tool, I had to decide on what programming language to use to develop this tool. One of the advantages of goal directed languages, like Unicon, is the simplicity of code in performing complex tasks. ...

In developing the tool, the project was divided into three sub subprojects

1. **Reading:** reading the YACC/Bison .y file, scanning it for tokens, identifying terminals and non terminal tokens and extracting the raw grammar.

2. **Analyzing:**

3. **Generating:**

4. **Testing:**

I had to break the project down into its component parts: reading of YACC/Bison .y files, creating a data structure for the context free grammar, generating a set of states for the automata for the grammar, generating a set of example syntax error fragments and testing each error fragment check its coverage.

Reading source files

The tool will be applied to the Unicon grammar, and to test generality, also the RTL language grammar from the Unicon implementation. For the Unicon grammar, the student will construct error diagnostic messages for each fragment in consultation with the instructor.

The directed study will be completed with an electronic submission of the stub generator and generated fragment files, and a written report that describes the tool and its use.

2 Background

3 Methods

4 Results

4.1 Neutral Exploration

(similar format as Baseline)

4.2 Breadth

(similar format as Baseline)

4.3 Depth

(similar format as Baseline)

4.4 Flat

(similar format as Baseline)

4.5 summation of results

5 Conclusions and future work

5.1 structural issues

5.1.1 DAG vs normal

5.2 computational issues

5.2.1

5.2.2

5.3

5.4

5.5 When most effective

5.6 Discussion

5.7 future work