

Racket-CFG

Author: David Benoit

Purpose

The goal of this library is to create a simulation of the following in Racket: 1) Context-Free Grammars 2) Push-Down Automata 3) Earley Parser for efficient Context-Free Grammar parsing

Why?

I wrote a library that is a music theory API to Racket's Rsound. As part of that library, I wrote a simple psuedo-random music generator, however, all of the valid chord progressions that it uses must be written out by hand.

This library will ultimately be used with racket-synthesis to create a wider variety of harmonic progressions for the generator, using Context-Free Grammars to specify the procedural generation of the harmonic progressions.

It will also allow the racket-synthesis users the opportunity to create their own progression generators and checkers.

Current Status

The project is currently unfinished. It is a larger project than I originally anticiated, and is taking longer than originally expected.

What is done so far: * Structure and organization of PDA data * Pattern matching for PDA state data * Structure and organization of CFG data * Some structure and organization of Earley parser data

What needs to be done: * PDA machine accepts * CFG accepts (via Earley Parser) * PDA \leftrightarrow CFG conversion

Run the PDA tests

Running the tests in a racket shell will produce cleaner output than using the Makefile or simply executing the file from the command line.

```
$ cd $PROJECT_DIR && make
```

OR

```
$ cd $PROJECT_DIR && racket
```

```
> (require "./cfg-pda/pda.rkt")
```