TigerWorks

Building and Running

Assuming you have the antlr jar in your classpath, you can build with the following commands:

```
java org.antlr.Tool Tiger.g
javac TigerCompiler.java
```

The first step should generate the Lexer and the Parser from the grammar file.
The second step builds the “driver” program, or what you will run to actually parse a given tiger program.

To run the driver program, run the following commands:

```
java TigerCompiler <filename>
```

If your program contains no syntactic errors, it will output a success message to stdout. Additionally, it will generate a DOT file named AST.dot. Assuming you have GraphViz installed, you can convert the DOT file into a PNG file by the following command:

```
dot -Tpng AST.dot -o <filename>.png
```

You can view this in any image viewer that supports the PNG format.

If your program is syntactically malformed, the parts of your program that do not meet the parser’s expectation will be printed to stdout, in addition to a potential reason explaining why it was not correct.

Code organization and layout

The sample programs and any ASTs of the syntactically correct programs will be found in the samples folder.

The grammar can be found in Tiger.g. This contains all lexer and parser rules, as well as all overridden methods for the parser and lexer. This is the root directory of the project. The grammar has been written to be LL(1) as specified in the project instructions. Where necessary, there were changes to the grammar to make it LL(1) and as a result is not exactly as written in the tiger specification document. Any particularly radical changes were documented in the grammar file as a comment.

The driver program TigerCompiler.java can also be found in the root directory of the project.

Output of sample programs

The following 4 programs have been supplied:

- matrix.tig
    - Multiplies two 5x5 matrices together, prints out the result.
- fact.tig
    - Calculates the factorial of 5.
- fib.tig
    - Calculates the 10th fibonacci number.
- fibBad.tig
    - This is a malformed version of fib.tig

matrix.tig, fact.tig, and fib.tig are all parsed correctly. The AST for each is contained in matrixAST.png, factAST.png, and fibAST.png respectively.

fibBad.tig does not parse correctly. When trying to parse it, it reports all the errors that make this program syntactically malformed to stdout. A syntax tree is not generated for this program.