

CS 335A: Compiler Design

Assignment-1

Ayush Sekhari (12185)

Vasu Sharma (12785)

Margaux Dorido

January 30, 2015

1 Lexer Design

1.1 Lexer specifications

We have the following language specifications for our Lexer:

Source Language: ADA

Target Language: MIPS

Implementation Language: Python

Pythons Pylex library was used to build the Lexer

1.2 Building the Lexer

The Lexer has been implemented in python and can be executed as:

1. Change directory to the directory containing lexer.py
2. python ./lexer.py jada program name.adb

example: python ./lexer.py test1.adb

1.3 Language Constraints

Our Lexer doesn't handle the following cases:

1. Operator overloading is disallowed by our Lexer due to ambiguity in classifying the redefined operator as string or identifier as operator overloading requires the operator to be put in double quotes which makes it impossible to distinguish it from other strings.
2. Our Lexer doesn't allow user to use reserved types as identifier names for example I can't have an identifier named String or Integer etc. This has been done as it creates the ambiguity as whether to identify as a data type or an identifier.
3. Ada has some non-standard data types too which are defined within certain libraries. We are not creating separate token names for these non-standard data types. We have however handled standard data types like Integer, String etc.
4. In the representation of numbers in the exponential form, our lexer returns the value of a number in exponential form in the normal numeric decimal representation. This leads to overflow errors in case the exponential number is too large. example 19E200 is out of bounds of any data type in Ada and hence causes an overflow causing a garbage value to be returned. We haven't handled such issues.