CS335 Project
Milestone 1
Group ID - 31

Kartavya Damor(200492) Soruabh Mina(200996) Kevalkumar Solanki(200991)

Instructions

The following tools were used in the project :-

- Flex : flex 2.6.4
- Bison : bison (GNU Bison) 3.8.2
- Dot : dot - graphviz version 2.43.0 (0)
- g++: g++ (Ubuntu 11.3.0-1ubuntu1 22.04) 11.3.0

The MakeFile contains the compiling codes for the flex bison and c++ program,

```
compile:
  bison -dv parser.y
  flex scanner.l
  g++ -o ASTGenerator AST.cpp parser.tab.c lex.yy.c -lfl
```

We have implemented 4 commands that can be viewed by calling the help command:-

- ./ASTGenerator −−help

```
--help            Display available options.
--input <file>    take <file> as input file.
--output <file>   will print the dot script in <file>.
--verbose         display the procedure in bit details.
```

For the execution of the compiler, follow the given commands :-

- make
- ./ASTGenerator −−input ../tests/<TestFile Name>.java −−output <OutputFile.dot> −−verbose
- dot -Tpdf <OutputFile.dot> -o <FinalOutput>.pdf

The generated AST will be constructed as the output in the <FinalOutput>.pdf

Implementation level details
VariableModifier does not support the keyword 'final'
Modifier contains limited keywords (public, private, static)
Since there are C-type array declarations therefore, the return value of the function cannot be java style array. For example, "public int[8] fun()" will not work.
The size of the abstract syntax tree is made wide for the sake of better understanding via the tree.
To reduce conflicts, we have further simplified the grammar to support only basic for loop
Some keywords like "enum", "record" is not supported.

Additional Features Implementations
We have added functionalities for import statements, i.e import statements like import java.util.*
Primitive type cast expressions are supported.
Support for Strings, including a few operations like concatenation, support for printing with println()

References

1. Java Language Specifications: https://docs.oracle.com/javase/specs/jls/se17/html/jls-19.html
2. https://github.com/mohitmo/CS335-Project (learnt Dot file generation)
3. A.Aho, R.Sethi, and J.Ullman. Compilers: Principles, Techniques, and Tools, 1st edition.
4. Testcases code generated from GFG, javaviz etc websites.