MILESTONE 1

MEMBERS

- ROHIT (200813)
- GAVISH GARG (200385)
- BHAVYA GARG (200270)

TOOLS USED

- Graphviz to convert .dot extension into .ps or .png extension
- Yacc for generating parser in LALR(1) grammar
- Lex for generating parser for generate **TOKEN** and **LEXEMES**

IMPLEMENTED FEATURES

- Primitive data types (int, long, float, double, and boolean)
- Multidimensional (max 3D) arrays.
- Arithmetic operators: +, -, *, /, %, ++, –
- Bitwise operators: &, |, ˆ, ˜, ≪, ≫, ≫
- Logical operators: &&, ||, !
- Assignment operators: =, +=, -=, *=, /=, &=
- Ternary operator
- Ignored the <>(diamond) operator
- Control flow using if-else, for, and while loops
- Methods and method calls, including both static and non-static methods
- Support for recursion
- Support for classes and objects. For class definitions, support public and private access modifiers, ignored default and protected.
- *Support for nested/inner classes is also supported.* This has been shown in test cases.

AVOIDED FEATURES

- Switch statements and do-while loops
- Interfaces
- Import modules
- Type parameters and Type Arguments.
- TRY and CATCH statements
- STRINGS

HOW TO RUN ?

The parser is generated using “—input” and “—output” as given below.

```
$ ./generateAST.sh --input <input file> --output <output file>
```

It also has “clean” and “—help” options that can be used to clean and help purpose respectively.

They are executed as

```
$ ./generateAST.sh clean
```

```
$ ./generateAST.sh --help
```

Moreover , there is also an option for “—verbose” which can be used along with the execute instruction to generate parser.output file.

```
$ ./generateAST.sh --input <input file> --output <output file> --verbose
```

The default input and output files are “input.txt” and “out.dot” respectively.