CS335 : Milestone 1

Arpit Kumar Rai (200190)
Avi Kumar (200229) Kunwar Preet Singh (200536)

March 2023

Tools and Utilities used

- Flex is used for our lexer, which is integrated with the parser. It returns tokens.
- Bison is used for implementing the parser. Official grammar is rewritten in a way that there is no conflict in lr(1) parsing.
- Make utility is used for automatic tracking for files and compilation.
- 3ac is generated using semantic actions.

Features implemented

- We have implemented all the basic features in the description.
- These include features like support for primitive data types, multidimensional arrays, basic operators (arithmetic, increment, relational , bitwise ...), control flow statements (if-else, for, while) etc.
- We have included various semantic checks as well like scoping errors, type errors, signature mismatch, initialization errors etc.

Instructions for Compilation

- Please change directory into *./milestone2/src/*
- Execute *make*
- Execute *./a.out* < *./path/input.txt* to give input.

Output Format

- symboltable is stored in such a manner that initial entries in the csv denote the symbol table entries, for these the fields of ScopeINFO are absent.
- Entries with ScopeINFO filled denote the parent-child hierarchy of the scopes.

Instructions for Running test cases

- We have provided 10 non-trival programs that may be compiled using the parser.
- We have provided an easy python script to automate the testing of these testcases.
- Please change directory into *./milestone2/*
- Execute *python run.py*
- The tests are named *./tests/test_[1-10].java*, corresponding outputs include *./outputs/3ac[1-10].dot* and *./outputs/symtable[1-10].csv*.