CS335A Compiler Project Milestone 2

Aditya Kumar (210060)   Antriksh Gupta (210159)   Geetika (210392)

March 2024

Instructions

Use the `Makefile` provide in the `src` directory to compile the parser.
`milestone2/src$ make`

It produces an executable `./parser`. Now this executable can be used along with flags to parse an input program and generate the corresponding Three AC IR and the CSVs for symbol tables of each class, function and global symbol table. Following is the usage of the flags.

```
Usage:  ./parser <flags>
--input <input_file_path>:  Specify the input program file
--output <output_file_path>:  Specify the output three AC IR file
--verbose:  Generate additional details about parsing in "verbose.log" file
--help:  Prints information about flags to help the tester.  The output is printed in standard output.
```

Running the `./parser` executable parses the provided input file, performs semantic analysis and type checking and produces the three AC IR file `3AC.txt` in the current directory if no output file is provided, otherwise a file with the give name is generated. The CSVs for symbol tables are generated in the current directory. If input file is not specified then standard input is used.

Use the following command to clean up the executable `./parser`, the temporary files, the three AC IR file and the CSVs.
`milestone2/src$ make clean`

Additional features implemented:

- Function calls can be given as arguments to another Function. For e.g. `print(len(array))`

- Walrus operator is supported.

- L-value checking is supported.

1