

Milestone 3 - CS335A

Course : CS335A - Compiler Design

Indian Institute of Technology, Kanpur

Rajat Gattani : 210813, Mohak Singh Rana : 210614, Dhruv Garg : 210339

April 21, 2024

Tools and Utilities used

- The lexer is written in Flex and it returns the tokens to the parser
- Bison is used for implementing the parser. Official grammar is rewritten by removing ambiguous and irrelevant rules such that there are no conflicts in LALR(1) parsing
- 3AC generated as *.txt* file and symbol table generated as *.csv* file
- Final x86 code generated as *.s* file
- Make utility is used for automatic tracking for files and compilation

Features Implemented

- Complete support for normal or nested for/while loop with extensive support for break and continue statement
- Support for all conditional statements (if,elif and else)
- Support for functions including recursion as well as built-in functions like print, len and range which also support overloading among themselves like *range(len(data))* and *print(len(data))*.
- Support for classes and methods, including multilevel inheritance and constructors with support for self object along with `__init__` call withing class
- Error handling which raises error for:
 - **Type mismatch** : Type mismatch in assignment, Function predicted vs actual return type, formal vs actual parameters raises error
 - Number of parameters mismatch error for function
 - Use before declaration or re-declaration errors
- Complete support for strings including relational operators (`==`, `>`, `<`, `>=`, `<=`, `!=`) , overwriting, passing as parameter in functions and support for basic string concatenation like `x+y`, `x+"hello"` or `"hi"+"hello"`

Modifications in 3AC

- Added type of function in the 3AC instruction for function call
- *Heapalloc* for memory allocation for arrays using *malloc* in ASM
- *printstr* for handling printing of strings separately

Instructions for Compilation

- Please change directory into `./milestone3/scripts/`
- Execute: `./run.sh filepath` (path should be relative to current directory)
- The generated output is displayed on terminal and the corresponding 3AC, symbol table and ASM code for the given python code are generated in the outputs folder

Instructions for running testcases

- We have provided 5 non-trivial programs for compilation using the given compiler
- We have provided a bash script to automate the testing of these testcases
- Please change directory into `./milestone3/scripts/`
- Execute `./checking_script.sh`
- The tests are named `./tests/test [1-5].py` , corresponding outputs generated by code after compilation will be displayed on the terminal with respective labels. The 3AC, CSV and ASM files will be generated in respective folders of `./outputs/` for each of `test[1-5]`

Effort Distribution

Equal contribution by all the three members