

CSE304 Homework #1

Chaeyoung Lee
Jeonghoon Kim

2016-02-15

1 Overall

SSM interpreter is an interpreter that understands stack machine-based assembly language, having stack and store as memory areas. We implemented stack and store using data type list in Python library. All inputs and outputs are done by stdin (standard input) and stdout (standard output). Reading and parsing inputs are done by ssm.py file. Executing commands are done by instruction.py file. After execution, the program will print the top-most value from the stack and terminate.

2 File Structure

```
HW1
├── ssm.py: Including Main Function
└── instruction.py
```

3 Details

3.1 Main function

Main function (main) reads inputs from stdin by call scanner function. After reading all inputs, input validation is done by semanticanalysis function. If there is no error in the given input, our interpreter computes the input code by calling compile function in instruction.py. After executing, it will print the top-most value in the stack through stdout. If there is any error(s) occurred from any step, the program stops immediately with printing the error message and terminates.

3.2 Scanner function

Scanner function is implemented in ssm.py to get inputs from stdin. If there is no input, the program will be terminated immediately and print the error message. If not, it will go through the input line by line and parse each command with following argument if the command requires a argument. If # is detected, all characters after # are ignored and go to the next line. All parsed words are stored as array.

3.3 Semanticanalysis function

Semanticanalysis function is implemented in ssm.py to analysis inputs from scanner. This function checks every command and check if they have proper arguments. If there is any input command that is not one of given commands(ildc, iadd, isub, imul, idiv, imod, pop, dup, swap, jz, jnz, jmp, load, store), the program terminates immediately with error message. If the command is ildc and the following input is not a integer number, the program will be terminated. For jump instructions, if there is no proper label in the code, the program will be terminated.

3.4 Compile function

Compile function in instruction.py executes commands from input. Stack is defined as a list, named stack. Store has two list arrays; store_addr and store_value. Store_addr array holds addresses of each stored value and store_value holds the value. Store_addr and store_value treat like a pair by index; for example, store_addr[0] will have the address of store_value[0], vice versa. If there is any error while executing, the program will be terminated immediately with printing proper error message. Compile function returns the topmost value from the stack.