Instructor: Gholamreza Ghassem-Sani	40 - 414 – Compiler Design (spring 1401) **User manual of the three-address codes Tester Program**[1]	Department of Computer Engineering Sharif University of Technology

The tester program has been designed to test the three address codes produced by the compilers which have been implemented as the programming project of the Compilers course. Therefore, it is necessary that the codes produced by these compilers to have identical formats according to the following explanation:

In order to get the desired result, each three address code must be exactly in the following format:

0 (JP, 9, ,)

where there is a line number in the beginning followed by a tab (\t) character and a three address code.

- Note that the tester program itself assigns a line number to you codes and the line numbers given by you are just to make the three address program easier to trace by the user.
- There must be exactly one Tab character between the line number and the three address code.
- There must be at least one space between the components of each three address code. You must not use a Tab character for this purpose. In the above example, there is a space between JP and 9.
- Extra commas for missing operands are not important. However, it is recommended that these commas to be added.
- Operators of the three address codes must have correct spelling (according to the project definition), and must be typed in all capital letters.
- To transfer the control to line L by JP or JPF, DO NOT use a "#" character (i.e., used for immediate values) before the line number L. The line number should be used directly.
- For an indirect jumping to line number L, there must be first an ASSIGN three address code to store #L in a memory location, say T, and then there could be an indirect jump to L by using @T.
- In the case of referring to a memory location without any value, the tester program stops and reports the error.
- The produced codes must be saved in a text file called 'output.txt' and located in the same folder as the tester program.
- When tester runs a three address program, it shows the order in which the three address codes are executed. Moreover, the values that are to be printed in the standard output are represented by a PRINT expression.

[1] The tester program has been implemented by Romina Jafarian, at Computer Engineering Department of Sharif University of Technology (Tehran) in Fall 2017.
It has later been improved by Nazanin Azarian and Amirreza Mirzaei, at Computer Engineering Department of Sharif University of Technology (Tehran) in Spring 2022.

- While running the tester, if you pass in -d flag, the program will proceed line by line which is suitable for debugging.
- On the last line of output, the total memory that your compiler has used is printed out.

The following figures show two samples of running a three address program by the tester program (i.e., called “a.out” in these figures) and the outputs of the tester program:

```
~/Desktop/test ./tester
--->  PC = 0    command : (ASSIGN, #0, 500, )
--->  PC = 1    command : (ASSIGN, #0, 504, )
--->  PC = 2    command : (ASSIGN, #41, 508, )
--->  PC = 3    command : (ASSIGN, #516, 556, )
--->  PC = 4    command : (ASSIGN, #0, 516, )
--->  PC = 5    command : (ASSIGN, #0, 520, )
--->  PC = 6    command : (ASSIGN, #0, 524, )
--->  PC = 7    command : (ASSIGN, #0, 528, )
--->  PC = 8    command : (ASSIGN, #0, 532, )
--->  PC = 9    command : (ASSIGN, #0, 536, )
--->  PC = 10   command : (ASSIGN, #0, 540, )
--->  PC = 11   command : (ASSIGN, #0, 544, )
--->  PC = 12   command : (ASSIGN, #0, 548, )
--->  PC = 13   command : (ASSIGN, #0, 552, )
--->  PC = 14   command : (ASSIGN, #0, 560, )
--->  PC = 15   command : (ASSIGN, #0, 564, )
--->  PC = 16   command : (ASSIGN, #1, 560, )
--->  PC = 17   command : (MULT, #4, #0, 1000)
--->  PC = 18   command : (ADD, 1000, 556, 1004)
--->  PC = 19   command : (ASSIGN, #5, 564, )
--->  PC = 20   command : (ASSIGN, 564, 560, )
--->  PC = 21   command : (ASSIGN, 560, @1004, )
--->  PC = 22   command : (ASSIGN, 560, 500, )
--->  PC = 23   command : (PRINT, 500, , )
PRINT    5
--->  PC = 24   command : (MULT, #4, #0, 1008)
--->  PC = 25   command : (ADD, 1008, 556, 1012)
--->  PC = 26   command : (ASSIGN, @1012, 500, )
--->  PC = 27   command : (PRINT, 500, , )
PRINT    5
--->  PC = 28   command : (ASSIGN, 564, 560, )
--->  PC = 29   command : (MULT, #4, 560, 1016)
--->  PC = 30   command : (ADD, 1016, 556, 1020)
--->  PC = 31   command : (ASSIGN, #7, 564, )
--->  PC = 32   command : (ASSIGN, 564, @1020, )
--->  PC = 33   command : (ASSIGN, 560, 500, )
--->  PC = 34   command : (PRINT, 500, , )
PRINT    5
--->  PC = 35   command : (ASSIGN, 564, 500, )
--->  PC = 36   command : (PRINT, 500, , )
PRINT    7
--->  PC = 37   command : (MULT, #4, #5, 1024)
--->  PC = 38   command : (ADD, 1024, 556, 1028)
--->  PC = 39   command : (ASSIGN, @1028, 500, )
--->  PC = 40   command : (PRINT, 500, , )
PRINT    7
Total memory used: 24
```

```
~/Desktop/test ./tester -d
--->  PC = 0    command : (ASSIGN, #0, 500, )

--->  PC = 1    command : (ASSIGN, #0, 504, )

--->  PC = 2    command : (ASSIGN, #41, 508, )

--->  PC = 3    command : (ASSIGN, #516, 556, )

--->  PC = 4    command : (ASSIGN, #0, 516, )

--->  PC = 5    command : (ASSIGN, #0, 520, )

--->  PC = 6    command : (ASSIGN, #0, 524, )

--->  PC = 7    command : (ASSIGN, #0, 528, )

--->  PC = 8    command : (ASSIGN, #0, 532, )

--->  PC = 9    command : (ASSIGN, #0, 536, )

--->  PC = 10   command : (ASSIGN, #0, 540, )

--->  PC = 11   command : (ASSIGN, #0, 544, )

--->  PC = 12   command : (ASSIGN, #0, 548, )

--->  PC = 13   command : (ASSIGN, #0, 552, )

--->  PC = 14   command : (ASSIGN, #0, 560, )

--->  PC = 15   command : (ASSIGN, #0, 564, )

--->  PC = 16   command : (ASSIGN, #1, 560, )

--->  PC = 17   command : (MULT, #4, #0, 1000)

--->  PC = 18   command : (ADD, 1000, 556, 1004)

--->  PC = 19   command : (ASSIGN, #5, 564, )

--->  PC = 20   command : (ASSIGN, 564, 560, )

--->  PC = 21   command : (ASSIGN, 560, @1004, )

--->  PC = 22   command : (ASSIGN, 560, 500, )

--->  PC = 23   command : (PRINT, 500, , )
PRINT    5
```