Cover Page
CS323 Programming Assignments

1. Names [1. Omar Al Nabulsi]

2. Assignment Number [3]

3. Due Dates **Softcopy** [12/10], **Hardcopy** [12/11]

4. Turn-In Dates **Softcopy** [12/10], **Hardcopy** [12/11]

5. Executable FileName [executable]

(**A file that can be executed without compilation by the instructor**

6. LabRoom [CS-101]

(Execute your program in a lab in the CS building before submission)

7. Operating System [Linux]

To be filled out by the Instructor:

GRADE:

COMMENTS:

CS323 Documentation

1. Problem Statement:

   To write a semantic analyzer and assembly instructor. The syntax analyzer will be created using 29 syntax function rules given from project one and two. The assignment will consist of a symbol table handling and generate assembly code for the simplified version of Rat19F.

2. How to use the program in Linux:
   - Open terminal and run the following commands:
     - g++ executable
     - ./executable
     - Type test case file when asked name

     OR
   - Place execute.sh file on Desktop along with test case files for ease of use.
   - Open terminal and type the following command:
     - cd Desktop
     - sh execute.sh
     - Type test case file name when asked
     - Open output file to view further details such as lexeme, symbol table and instruction table

3. Design of the program:
   - Semantics considering that “true” has an integer value of 1 and “false” has an integer value of 0, no arithmetic operations allowed.
   - Symbol handling, each entry in the symbol holds a lexeme and memory address where identifier is placed within symbol table.
   - Check to see if the identifier is already in the table, print out all identifiers in the table.
   - Generating assembly code, add code to parser that will produce assembly code instructions kept in an array. Content is then printed out to produce listing of assembly code.

4. Limitations:
   - Test case must be completely free of syntax errors to run
   - Do not enter the character “%” in test case, the program will not run

5. Shortcomings:
   - None