

50 pts

Name: _____

Class Day / Time: _____

Due Date: _____

Assignment #4 – Assembly - Implementing a Finite-State Machine

In this assignment you will create and implement a finite state machine to validate a sequence of numerical digits and letters as part of a string. The finite state machine will be implemented using assembly as discussed in class and the program will perform the following steps:

- 1) Input from the console a string from the user and store it in a memory location labeled *inputStr*.
- 2) Use the state machine to validate the input string and output to the console whether or not the input string is valid.

The following would be the valid sequence for the input string:

- a) one numerical digit ('0' ... '9') followed by
- b) one or more lower-case letters ('a' ... 'z') followed by
- c) one or more numerical digits ('0' ... '9') followed by
- d) the exclamation mark '!'

For instance, the following would be valid sequences:

1abc0!

2zzzzzz55555!

7c2!

Draw the finite-state machine diagram, a flowchart and write the corresponding assembly language program in the flowchart for the program you are implementing. It is not necessary to indicate memory locations in the flowchart, but include **all the variable** declaration as part of the flow chart.

Implement the program; test the program a number of times with different strings. You will need to turn in **four** test runs for the program:

- a. A valid sequence including at least five letters
- b. A valid sequence including at least ten numerical digits.
- c. A valid sequence including at least three letters and five numerical digits.
- d. An invalid sequence.

Turn in (STAPLED IN THIS ORDER)

1. The **FIRST PAGE** of this assignment as a coversheet
2. The **finite-state machine** diagram to validate the sequence described in this assignment.
3. Include the **flowchart** properly documented. The listing of **.asm source code** properly documented.
4. The **four** output from the program, either pasted into .asm source code or using print screen.