

605.202.31

## Data Structures

### Lab1 – Stack

#### Choice 1

#### Problem

Assume a machine that has a single register and six instructions.

LD	A	Places the operand A into the register.
ST	A	Places the content of the register into the variable A.
AD	A	Adds the content of the variable A from the register.
SB	A	Subtracts the contents of the variable A from the register.
ML	A	Multiplies the contents of the register by the variable A.
DV	A	Divides the contents of the register by the variable A.

Write a program that accepts a postfix expression containing single letter operands and the operators +, -, \*, and / and prints a sequence of instructions to evaluate the expression and leave the result in the register. Use variables of the form TEMP as temporary variables. For example, using the postfix expression ABC\*+DE-/ should print the following:

```
LD    B
ML    C
ST    TEMP1
LD    A
AD    TEMP1
ST    TEMP2
LD    D
SB    E
ST    TEMP3
LD    TEMP2
DV    TEMP3
ST    TEMP4
```

Keep in mind that you are NOT evaluating the postfix expression, you are generating the machine language instructions that would perform the evaluation if actually executed.

In your analysis, be sure to discuss the implementation you choose and why, why a stack makes sense. Consider a recursive solution (you do not need to implement recursion) and compare it to your iterative solution. Is one better than the other? Why? Tell us what you learned and what you would do differently.

Be sure to review the programming assignment guidelines, including the formatting requirements for the analysis. You may not use library functions. You must write your own code, in particular you must write the stack code. Be sure to include the stack source code in your submission. You must read and write from named files.