**Assignment #4 (80 Points) – COSC 2336 – Dr. Leonard Brown**
**Due: November 17, 2017 (at 11:59pm)**

## General Description

There are multiple parts (I and II) to this assignment.

## Part I (15 points)

Answer the following questions.

1. If you enqueue the letters A, E, I, O, and U in order into a queue of characters and then dequeue them, in what order will they be removed from the queue?

2. What do the initially empty queues queue1 and queue2 contain (in order) after the following sequence of operations? What do the variables x, y, and z contain?

```
queue1.enqueue(10)
queue1.enqueue(20)
x = queue1.dequeue()
queue2.enqueue(30)
queue2.enqueue(40)
y = queue2.dequeue()
queue1.enqueue(50)
queue1.enqueue(60)
z = queue1.dequeue()
queue2.enqueue(x)
queue1.enqueue(y)
queue2.enqueue(z)
x = queue1.dequeue()
y = queue2.dequeue()
z = queue1.dequeue()
```

3. Explain how an array-based implementation of a queue can suffer from "rightward-drift". Why doesn't a reference-based implementation have this problem?

## Part II (65 points) – EZ Programming Language

Although the EZ language only has five keywords, it allows users to declare and call functions. The five keywords in the EZ language are as follows:

| | |
|---|---|
| `proc` <*identifier*> | declares a procedure named *identifier* |
| `call` <*identifier*> | calls procedure *identifier* |
| `stop` | ends execution of current procedure |
| `echo` <*text*> | displays lowercase text and the values of uppercase variables until of the end of the line followed by a new line |
| `copy` <var> <*in-expr*> | evaluates an infix expression consisting of uppercase variables, single digit values, addition signs, and multiplication signs and places the result in variable *var*. |

There are some restrictions on the function calls that are allowed in the EZ language.

- A function may call only functions that have previously been declared.
- The program begins executing the function named `main`. This must be the last function declared. The program terminates when a stop command is encountered in the main function.
- Variables are only allowed to be uppercase letters. Thus, no more than 26 variables can be used in a program. All should be initialized to 0.
- All variables are global.
- The echo command can only print lower case text. If an upper case character is in the echo command, the command should treat it as a variable and display its value.
- As in most languages, when one function calls another, execution is temporarily suspended in the first function until the called function terminates.

### Input/Output

Your program should read the input program from a text file. Each line of the input contains one statement of the EZ programming language. No commands use multiple lines. Your program should print the output that would be generated from the input EZ program.

### Sample Input

```
proc alpha
echo starting alpha
copy X X+Y
stop
proc beta
echo starting beta
copy Y Y*2
call alpha
copy Z Z+X
stop
proc gamma
echo starting gamma
copy Z Z+1
call beta
stop
proc main
copy X 1+2*3+4
copy Y 1+1
copy Z X*Y
echo the value of z starts at Z.
call alpha
call beta
call gamma
echo x is X and y is Y and z is Z.
stop
```

### Sample Output

```
the value of z starts at 22.
starting alpha
starting beta
starting alpha
starting gamma
starting beta
starting alpha
x is 25 and y is 8 and z is 65.
```

### Assumptions

- There are no blank lines in the input EZ program.
- The EZ program will be syntactically valid.
- EZ procedure declarations will not be nested.
- The length of each EZ procedure name will be between 1 and 10 letters, inclusive.
- All EZ commands will be given in lowercase.
- Arguments to the `echo` command will only contain letters or spaces
- The main class of your java program should be named **Assignment4**.
- The name of the file containing the input EZ program will be sent as a command-line argument (args[0]). So, this program will be executed with a line similar to the following: `java Assignment4 input.txt`

### Grading

The program in Part II will be graded based on the following criteria

| | |
|---|---|
| Implementation of Requirements | 25 points |
| Correct Computation of Input/Output | 25 points |
| Design and Documentation | 15 points |

### Submission

Submit your assignment through Canvas. If your assignment contains multiple files, zip them into a single folder before submitting.

### Notes

The same requirements in Assignment #1 regarding the documentation and formatting style apply to this assignment as well. Points can be deducted from your assignment based on the quality of its presentation. Handwritten assignments will **not** be accepted.