CS 218 – Assignment #6

Purpose: Become familiar with data conversion, addressing modes, and assembly language macro's.
Due: Wednesday (6/19)
Points: 75

Assignment:

Write an assembly language program to convert octal[1]/ASCII string to integers and integers to octal/ASCII strings. Using the provided main, the program has four steps are follows:

1. Write the code to convert a string of ASCII digits representing an octal value into an integer (double-word sized). This code should be placed in the provided main at the marked location (step #1) and will convert the string **`oNum1`** (octal/ASCII representation) into an integer stored in the variable **`iNum1`**.
2. Write the code to convert an integer into a string of ASCII digits representing the octal value (NULL terminated). This code should be placed in the provided main at the marked location (step #2) and will convert the integer stored in the variable **`iNum1`** into a string **`oNum1`** (octal/ASCII representation).
3. Convert the code from step #1 into a macro, ***octal2int***, which is called multiple times in the next part of the provided main.
4. Convert the code from step #2 into a macro, ***int2octal***, which is called multiple times in the next part of the provided main.

Write two macros to convert to perform octal/ASCII to decimal and decimal to octal/ASCII conversion. You may assume valid/correct data. As such, no error checking is required.

The provided main will also invoke a print string macro, which will display the strings to the screen. The print macro does not perform any error checking, so the data must be correct in order for the display to work. *Note*, since the program displays the results to the screen, typing the program name (without the debugger), will display the results to the screen.

Debugging

Since macro's can be difficult to debug, the code for steps 1 and 2 should be working before attempting steps 3 and 4.

The code for a macro will not be displayed in the source window. In order to see the macro code, display the machine code window (**View → Machine Code Window**). In the window, the machine code for the instructions are displayed. The step and next instructions will execute the entire macro. In order to execute the macro instructions, the **stepi** and **nexti** commands must be used.

To help check results, an on-line conversion is available at the following URL:
http://www.cleavebooks.co.uk/scol/calnumba.htm.

1 For more information regarding base-8 Octal representation, refer to: http://en.wikipedia.org/wiki/Octal

Submission:
When complete, submit:

- *A copy of the source file via the class web page (assignment submission link). Assignments received after the 1:00 PM will not be accepted.*

Example Output:
The results, as displayed to the screen, would be as follows:

```
ed@ed-vm% ./ast6
**********************************
CS 218 - Assignment #6


First Number: +12345
First Number (*2): +24712

----------------------
List Stats

List Sum:
-3756

List Average:
-773

. . .  truncated for space . . .
```

Assignment #6 - Data
Refer to the provided main for the provided data sets.

Debugger Commands:
Below is an example of some of the commands to display some of the variables within DDD for assignment #6. You will need to include the rest.

```
x/s &oNum1
x/dw &iNum1
x/4s &oLst1
x/dw &len1
x/dw &sum1
x/dw &ave1
```

Note, in DDD, you can select **View → Execution Window** to display a separate window that shows the output (which would normally be displayed to the screen).