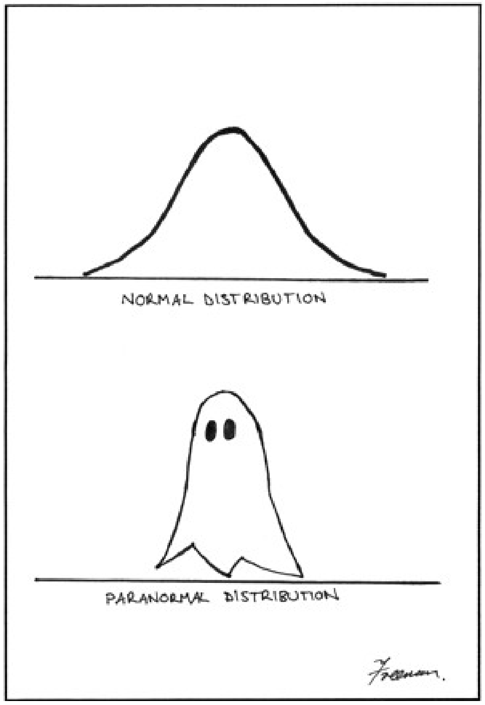
CS 218
Homework, Asst. #7

Purpose: Write a simple assembly language program to sort a list of numbers. Learn to use addressing modes, arithmetic operations, and control instructions.
Due: Tuesday (2/19)
Points: 100

Assignment:

Write a simple assembly language program to sort an array of numbers into ascending (small to large) order. Additionally, find the minimum, statistical median, maximum, sum, and average of the list. You should find the minimum and maximum after the list is sorted (i.e., min=array[0] and max=array[len-1]). For an odd number of items, the median value is defined as the middle value. For an even number of values, it is the integer average of the two middle values. The median must be determined *after* the list is sorted.



Use the following Comb[1] sort algorithm which sorts in ascending order:

```
void function combSort(array, length)
    gap = length
    swapped = true

    outer loop while gap>1 OR swapped = true
        // update gap for next comb sweep
        gap = (gap * 10) / 12
        if gap < 1
            gap = 1
        end if

        i = 0
        swapped = false

        inner loop until i + gap >= length     // single comb sweep
            if  array[i] > array[i+gap]
                swap(array[i], array[i+gap])
                swapped = true
            end if
            i = i + 1
        end inner loop
     end outer loop
end function
```

You ***must*** use the above comb sort algorithm above (i.e., do **not** use a different sort). *Note*, the algorithm assumes array index's start at 0. As necessary, you can define additional variables. ***Submissions not based on this algorithm will not be scored***.

The program should display the minimum, maximum, statistical median, sum, and average to the screen in dozenal (base 12) format. Use the provided main, which includes the print routines. You will need to add the ***int2dozenal*** macro code from the previous assignment.

[1] For more information, refer to: http://en.wikipedia.org/wiki/Comb_sort

All data must be treated as ***signed*** integers (i.e., positive and negative numbers). As such, the IMUL and IDIV instructions should be used (not the DIV and/or MUL). Do not change the provided data types/sizes.

The program should display the minimum, maximum, median, sum, and average to the screen in dozenal. Use the provided main, which includes the print routines. You will need to add the macro code from the previous assignment.

Submission:

When complete, submit:

- A copy of the **source file** via the class web page (assignment submission link). Assignments received after the start time of class (section 1, 2:30 PM or section 2, 5:30 PM) will not be accepted.

Data Declarations:

Refer to the provide main for the provided data declarations.
As necessary, you can define additional variables.

Integer to Dozenal Macro:

This assignment uses the integer to dozenal conversion macro from assignment #6. The provided main includes a place to cut-and-paste the code from the assignment #6 macro into the assignment #7 template. The macro is used, along with some other provided macro's, to display output to the screen (as shown below). The data must be correct, or at least very close to correct, in order for the final display to work correctly.

Example Output:

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

```
CS 218 - Assignment #7

Minimum:          -534X
Maximum:          +65X9
Median:            +4E5
Sum:            +19466X
Average:           +X31
```

Note, since this program displays output to the screen, it can be executed without the debugger.

Debugging Tips

- Use comments!!
- Follow the algorithm directly (do not attempt to optimize).
- Comment each part of the algorithm (so you can match the algorithm to the appropriate subset of code).
- Develop a debugger input file first (based on previous ones) carefully verifying the debugger commands based on the specific data types.
- You can temporarily change the array length to a smaller number (i.e., 5-10) for testing.