

## CS 218 – Assignment #8

Purpose: Learn assembly language functions. Additionally, become more familiar with program control instructions, function handling, and stacks.

Due: Monday (6/20)

Points: 150

### Assignment:

Write four simple assembly language functions to provide some statistical operations as described below. You will be provided a main function that calls the following functions (for each set of data).

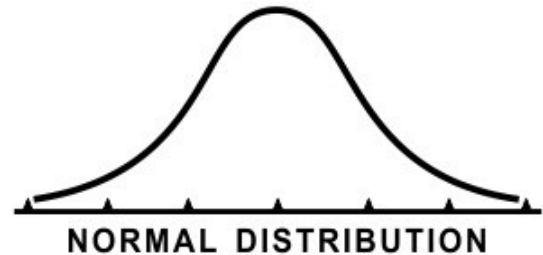
- Void function, **bubbleSort()**, to sort the passed array of numbers into descending order (large to small). You **must** use the bubble sort algorithm from assignment 7, modified to sort in descending order.
- Void function, **simpleStats()**, to find the minimum, median, maximum, count of even values, and count of values evenly divisible by 5 for a list of numbers. *Note*, 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.
- Value returning function, **iAverage()**, to compute and return the integer average for a list of numbers. A 32-bit integer function returns the result in *eax*.
- Void function, **lstStats()**, to compute and return the variance and standard deviation for a list of numbers. The formula for variance is as follows:

$$\text{variance} = \sum_{i=0}^{\text{len}-1} (\text{list}[i] - \text{average})^2$$

$$\text{stdDev} = \sqrt{\frac{\text{variance}}{\text{length}}}$$

The **lstStats()** function must call the **iAverage()** function to find the average and the **stdDeviation()** (provided) function to calculate the standard deviation. *Note*, due to the data sizes, the variance summation must be performed as a quad-word (using the **add** and **adc** instructions, as described in section 7.5.1.1 on the class text).

All data should be treated as *signed* integers (IMUL, IDIV, and CDQ instructions). The functions must be in a separate assembly file. The files will be assembled individually and linked together.



### **Submission:**

- All source files must assemble and execute on Ubuntu with **yasm**.
- Submit source files
  - Submit a copy of the program source file via the on-line submission.
  - Only the functions file (**ast8procs.asm**) will be submitted.
- Once you submit, the system will score the project and provide feedback.
  - If you do not get full score, you can (and should) correct and resubmit.
  - You can re-submit an unlimited number of times before the due date/time (at a maximum rate of 5 submissions per hour).
- Late submissions will be accepted for a period of 24 hours after the due date/time for any given lab. Late submissions will be subject to a ~2% reduction in points per an hour late. If you submit 1 minute - 1 hour late -2%, 1-2 hours late -4%, ... , 23-24 hours late -50%. This means after 24 hours late submissions will receive an automatic 0.

### **Program Header Block**

All source files must include your name, section number, assignment, NSHE number, and program description. The required format is as follows:

```
; Name: <your name>
; NSHE ID: <your id>
; Section: <section>
; Assignment: <assignment number>
; Description: <short description of program goes here>
```

Failure to include your name in this format will result in a loss of up to 5%.

### **Scoring Rubric**

Scoring will include functionality, code quality, and documentation. Below is a summary of the scoring rubric for this assignment.

Criteria	Weight	Summary
Assemble	-	Failure to assemble will result in a score of 0.
Program Header	3%	Must include header block in the required format (see above).
General Comments	7%	Must include an appropriate level of program documentation.
Program Functionality (and on-time)	90%	Program must meet the functional requirements as outlined in the assignment. Must be submitted on time for full score.

### Compile, Assemble, and Linking Instructions

You will be provided a main function that calls the functions. Your functions should be in a separate file (**ast8procs.asm**). The files will be assembled individually and linked together.

When assembling, and linking the files for assignment #8, use the provided **makefile** to assemble, and link. *Note, only* the functions file, **ast8procs.asm**, will be submitted. The submitted functions file will be assembled and linked with the provided main. As such, do not alter the provided main.

### Provided Data Sets:

Do not change the data types of the provided data. You may define additional variables as required.

```
; -----
; Data Sets for Assignment #8.

list1      dd      21,    27,    10,    22,    31
            dd      13,    12,    17,    19,    20
            dd      24,    11,    14,    30,    33
            dd      27,    34,    23,    37,    40
            dd      38,    18,    15,    25,    16
            dd      26,    39,    36,    50

len1       dd      29
min1       dd      0
med1       dd      0
max1       dd      0
evenCnt1   dd      0
fiveCnt1   dd      0
ave1       dd      0
var1       dq      0
```

The results for data set #1 are shown for reference:

```
List1:
$1 = {50, 40, 39, 38, 37, 36, 34, 33, 31, 30, 27, 27, 26, 25, 24,
23, 22, 21, 20, 19, 18, 17, 16, 15, 14, 13, 12, 11, 10}

len1: 29

Min1: 10
Med1: 24
Max1: 50
EvenCnt1: 15
FiveCnt1: 7
Ave1: 25

Var1: 2935
Std1: 10.06016384318486878
```