**Lab 01 Specification** – Implementing and Evaluating Algorithms
Due Friday, 26 Jan 2017
50 points

# Lab Goals

- Translate Algorithm to Program

- Use Std Draw in Java to plot experimental results

# Assignment Details

Firstly practicing how to transform algorithm into program written in Java. Secondly, get to know how to time the program execution through the program running time. Thirdly, using builtin functionality in Java to show experimental results on an applet.

## Random Generator

Write a method called **arrayGenerator** that randomly generate the values of the items inside any given array. The function needs to be generic so that it can be invoked at different times for different array sizes. For example:

- Dataset 1 contains 1 Million items.

- Dataset 2 contains 2 Million items.

- Dataset 3 contains 3 Million items.

- Dataset 4 contains 4 Million items.

- Dataset 5 contains 5 Million items.

- Dataset 6 contains 6 Million items.

- Dataset 7 contains 7 Million items.

- Dataset 8 contains 8 Million items.

- Dataset 9 contains 9 Million items.

- Dataset 10 contains 10 Million items.

Note: Each dataset correspond to an integer array in your implementation.

## Algorithm Implementation

Write a method called arrayMax that implement the following algorithm that we discussed in class to find the maximum element in any given array.

```
Algorithm arrayMax(A, n):
Input: An array A storing n Integers.
Output: The maximum element in A.
currentMax <- A[0]
for i <- 1 to n-1 do
   if currentMax < A[i] then currentMax <- A[i]
return currentMax
```

## Experimental Study

Write a function called functionTimer which would execute the arrayMax function and clock the running time for all the datasets generated by your program in the first step.
The function will then use **StdDraw** class available for you in the lab specification attachment to plot the running time for different datasets.
Note: X axis = Dataset ID [1, 2, 3, ... 10] Y axis = Running time of arrayMax function on those datasets.

# Submission Details

There is no submission needed. On or before the due date, you are responsible to demonstrate the correct working of your code by doing a detailed demo to either the Instructor. On instructors approval you are allowed to demo your work to the TA. The points for the lab will be based on the evaluations done on your demo and the working of your code. In addition, the feedback on your lab will be given after your demo.