### Portfolio Function Title: MaxHeap

Version number: 1

### Function Description:

Allows for a user to build a MaxHeap of objects with a specified size. Once a MaxHeap is built a user can add elements until the MaxHeap reaches the specified size. Once the MaxHeap is populated, the user can look at and remove the maximum element (according to the element type’s compareTo() method) in the MaxHeap.

### Author:

Nathan Mankovich

### Date Written/Last Modified:

05-06-2015 / 05-07-2015

### How to use the Function:

To use the stack program just instantiate a MaxHeap<obj> with an object of your choice and then you can use the different methods in your java code.

The program is stored in the portfolio.jar file, put that file in the same folder as YourProgram.java.

Open a Windows command window and enter:

To compile the program with YourProgram.java enter:

javac -cp .;portfolio.jar YourProgram.java

pause

Then, to run the program with the provided sample-mailing-list.csv enter:

java -cp .;portfolio.jar YourProgram

pause

User Interface: Java code and command line only.

### How the Function works:

The MaxHeap relies on the array data structure to store the elements in the heap.

When the MaxHeap is initially created it has a maximum size specified by the user, an initial size of 0 and has elements stored in the itms array of the user-specified maximum size.

The array holds all the elements, but the concept is that each element can have up to two children, a right child and a left child. The rule is that every parent in the MaxHeap is greater than its child.

### Supported Methods (including Inputs, Outputs, Features and Results by method):

* GetMax gets the maximum element in the MaxHeap
  + Input: none
  + Output: the maximum element in the MaxHeap
* RemoveMax removes the maximum element in the MaxHeap
  + Input: none
  + Output: the old maximum element in the MaxHeap
* Add adds an element to the MaxHeap
  + Input: the element to be added to the MaxHeap
  + Output: none

### Known problems and limitations:

This MaxHeap implementation does not allow the user to access the number of elements in the MaxHeap.

It would be nice to have a MaxHeap that is dynamically sizable instead of this statically sized array implementation. Thus, no elements can be added after the MaxHeap is full.

There is not a RemoveMin method because the minimum donor is difficult to find in a MaxHeap.