COSC 3100 – Data Structures II

Assignment 7
Deadline October 30, 2023

1) Develop an implementation of the template PriorityQueue class. The data should be stored as a 'heap' in an array within the class. This class should have the following public functions defined:

   PriorityQueue();
   Constructor to create an empty priority queue by dynamically allocation an array initially of size 12.

   ~PriorityQuene();
   Destructor to deallocate the dynamically allocated array.

   void enqueue(const T& value);
   Function to add an item to the priority queue. If the array is already fully occupied, then it should be expanded by an additional 10 elements.

   bool dequeue(T& value);
   Function to remove the item with the highest priority currently in the queue. If the array has 10 empty elements after this removal, then the size should be decreased by 10 elements, but should not go below the minimum array size of 12.

   void display() const;
   Function to display all information currently in the priority queue.

   void heapify(int i);
   Function to re-structure the priority queue following an item being removed.

2)
   a) Develop a program to store Stock objects in a Priority Queue. A minimum of 15 stocks should be stored in a data file named "Stock.txt". The data file will be read, and each Stock object added to the priority queue.

   b) The "priority" of a Stock should be based on the 'price'. The higher the price, the higher the priority. *This will require a small update to the existing Stock class.*

   c) Once all Stock objects have been inserted into the priority queue, they should all be displayed.

   d) Five Stock objects should then be removed from the priority queue. When a Stock is being removed it should be displayed. Also, the whole priority queue should be displayed after each removal.

THE DEPARTMENT STANDARDS FOR "STYLE GUIDELINES" SHOULD BE FOLLOWED IN ALL CODE.