Implement a Max Heap

1) Create a `Templated Heap Class` with the following member functions: (Get it to work with integers first)

`insert()`: Inserts a new object into the Heap. You can use a STL Vector to store the heap items. The method must be able to allow the vector to grow as the data size grows.

`remove()`: Removes the next element in the heap.

`trickleUp()`: Used to move a node from the last position to its correct position to satisfy the Heap condition. This should be an iterative implementation.

`trickleDown()`: Used to move a node from the top position to its correct position to satisfy the Heap condition. This should be an iterative implementation.

`display()` : Displays neatly on the screen all the elements in the heap.

Implement Heap Sort

2a) Create a `Templated HeapSort function` that uses the Heap Class.

2b) Write a main to test your `HeapSort function` on a vector of 1000 random integers. This main function also should include a timer that displays on screen the run time of the Heap Sort.

Recursive Implementations

3a) Now provide `trickleUpRec()` and `trickleDownRec()` which are recursive implementations of `trickleUp()` and `trickleDown()`.

3b) What is the time difference between the recursive and the iterative implementations?