CE 242 - Data Structures and Algorithms
Spring 2009
Lab Assignment 6

1. Among many different sorting algorithms, heap sort is an interesting one due to its use of an array implementation of a data structure called “Heap” which is also used to implement some other important data structures such as “Priority Queues”.

Heap sort algorithm can be implemented as a function that takes an array and its size as input and gives the in-memory sorted array. A C++ implementation for heap sort that only sorts string arrays is provided with the assignment as `"HeapSortStr.h"` and `"HeapSortStr.cpp"` source files. What you are required to do is to implement the `heapify()` function whose prototype declaration is as follows:

```
static void heapify(char **pArr, int rootIdx, int endIdx);
```

The function `heapify()` takes an input array (which is formatted as a heap), index of the root node of the heap and the last node’s index. It should treat the root node as it does not satisfy the “heap property” and should try to swap it with its biggest child if the biggest child is bigger than the root. If a swap occurs between the root and its children, this process should continue recursively until all nodes satisfies the “heap property” in other words until the root node is moved to a proper location according to the “heap property”.

For more information about the heap data structure and the heap sort algorithm check these links:
http://en.wikipedia.org/wiki/Heap_(data_structure)
http://en.wikipedia.org/wiki/Heapsort