CIS 121  
Homework Assignment 6

Given: February 18, 2016  
Due: February 26, 2016

Note: The homework is due electronically on the course website on Friday, February 26 by 10:30 am. For late submissions, please refer to the Late Submission Policy on the course webpage: http://www.seas.upenn.edu/~cis121

You must use the hw121.cls template provided on the course website.

Please write concise and clear solutions; you will get only a partial credit for correct solutions that are either unnecessarily long or not clear.

You are allowed to discuss ideas for solving homework problems in groups of up to 3 people but you must write your solutions independently. Also, you must write on your homework the names of the people with whom you discussed.

Finally, you are not allowed to use any material outside of the class notes and the textbook. Any violation of this policy may seriously affect your grade in the class.

1. Show that, for any \( n \), there is a sequence of insertions in a max-heap that requires \( \Omega(n \log n) \) time to process.

2. Consider a max-heap \( T \) for storing \( n \) keys. Give an efficient algorithm for reporting all the keys in \( T \) that are greater than or equal to a given query key \( x \) (which is not necessarily in \( T \)). Your algorithm should run in \( O(k) \) time, where \( k \) is the number of keys reported.

3. A Staq is a data structure combining properties of both stacks and queues. It can be viewed as a list of elements written left to right such that three operations are possible:
   
   • \textsc{StaqPush}(x): add a new item \( x \) to the left end of the list.
   
   • \textsc{StaqPop}(): remove and return the item on the left end of the list.
   
   • \textsc{StaqPull}(): remove and return the item on the right end of the list.

   Implement a Staq using three stacks and \( O(1) \) additional memory so that the amortized time for any \textsc{StaqPush}, \textsc{StaqPop}, or \textsc{StaqPull} operation is \( O(1) \). In particular, each element in the Staq must be stored in exactly one of the three stacks. Again, you are only allowed to access the component stacks through the interface functions \textsc{Push} and \textsc{Pop}.

4. In class we studied binary Huffman coding of the letters in the alphabet \( S \). Generalize Huffman’s algorithm to create a ternary encoding of the letters in \( S \).
5. (a) Using Huffman encoding scheme on a set $S$ of $n$ symbols with frequencies $f_1, f_2, \ldots, f_n$, what is the longest a codeword could possibly be? Give an example set of frequencies that would produce this case.

(b) Prove that if some character occurs with a frequency more than $2/5$, then there is guaranteed to be a codeword of length 1.

(c) If all characters occur with frequency less than $1/3$, then there is guaranteed to be no codeword of length 1.