**These problems will help you understand the sorting. This Homework should prepare you for Project 2, and 3**

**Problem 1**.[20 points] *(IndirectSort.java)* Implement the static method sort() in IndirectSort.java that indirectly sorts a[] using insertion sort, ie, not by rearranging a[], but by returning an array perm[]  such that perm[i] is the index of the ith smallest entry in a[].

$ java IndirectSort

I N D I R E C T I N S E R T I O N S O R T E X A M P L E

<ctrl -d>

A C D E E E E I I I I L M N N N O O P R R R S S T T T X

**Problem 2**. .[20 points] *(MergeQueues.java)* Implement the static method merge() in that takes two queues of sorted items as arguments and returns a queue that results from merging the queues into sorted order. Your implementation must be linear and must not alter the input queues.

$ java MergeQueues

A B C D E F G H I J K L M N O P Q R S T U V W X Y Z

**Problem 3.** .[20 points] (*CertifyHeap.java*) Implement the static method maxOrderedHeap() in CertifyHeap.java  that takes an array a[]  of Comparable  objects and returns true  if a[] represents a maximum-ordered heap and false  otherwise. Your implementation must be linear.

$ java CertifyHeap

0 T H R P S O A E I N G

<ctrl -d>

false

$ java CertifyHeap

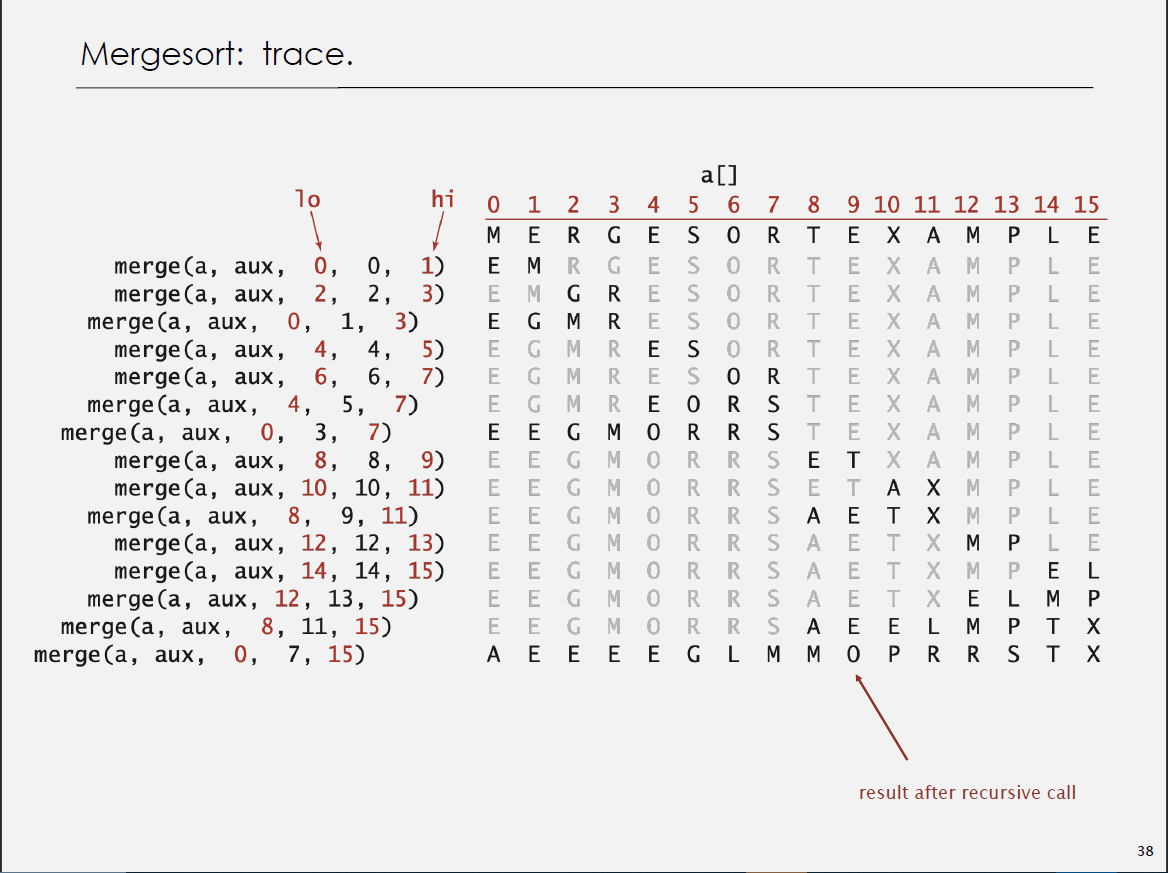
0 T S R P N O A E I H G

<ctrl -d>

true

**Problem 4.** .[20 points]Like slide 39, trace the merge sort for input “E A S Y Q U E S T I O N”

Below is slide for your reference:



Problem 5.[5 points]: About how many compares will Quick sort make when sorting an array of N items that are all equal?

Problem 6.[5 points]: Suppose that the sequence

|  |
| --- |
| P R I \* O R \* I \* T \* Y \* \* \* Q U \* E \* \* \* U \* E |

(where a letter means *insert* and an asterisk means *remove the maximum*) is applied to an initially empty priority queue. Give the sequence of values returned by *remove the maximum* operations.

Problem 7.[10 points]: Explain in brief different ways of heap construction

**Submitting Information:**

* DO NOT post your code on Piazza
* Use the code I provided for each problem. DO NOT delete any function
* Submit your work on Canvas.
* DO NOT change the name of .java files.
* Put all the .java files in one zip file and name it <last><first>.zip
* The deadline is March 19th at 11:59PM
* Follow the guidelines in homework rubric 3