Homework Assignment 8

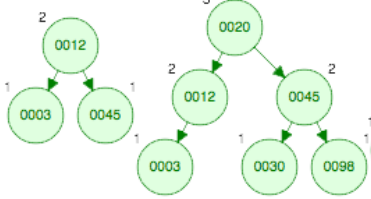
Due date: November 24th, 11:55pm EST

Problem 1

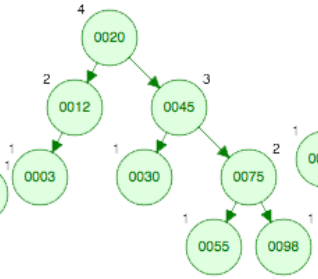
Node that caused change:

3

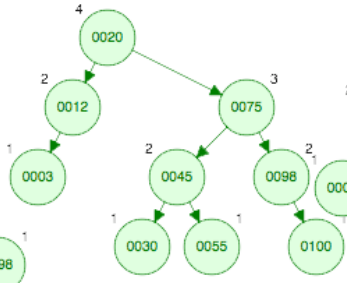
What tree looks like after rotation:



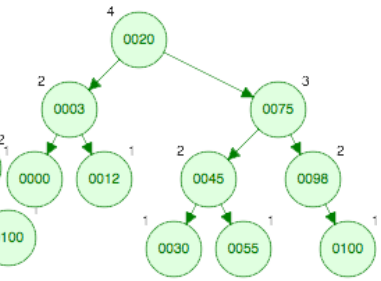
30



55

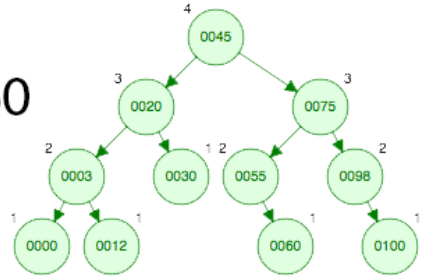


100



0

60

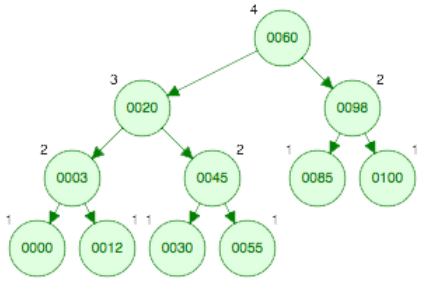


Problem 2

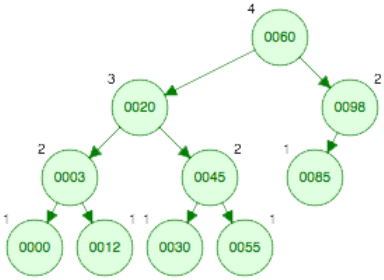
Node that was removed:

75

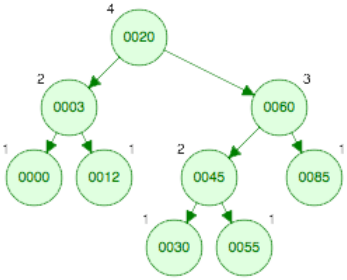
What tree looks like after node is removed:



100



98



Problem 3

A: There are 7 constructors inside the PriorityQueue Class

B: The array in which data is stored is declared on line 97

C: The initial capacity is 11, and the rules for growing the array are to double if the array size is small, otherwise to grow by 50%

D: methods called in order of appearance when adding to the Queue: offer(), grow(), possibly siftUp(), depending on whether i==0 or not. If siftUp is called, it will either call siftUpComparable, or siftUpUsingComparator, depending on whether Comparable == null.

E: No methods called when using the poll() method, unless s != 0, in which case siftDown() is called, which would subsequently call the siftDownUsingComparator or siftDownComparable, depending on whether Comparable == null.

F: line 652 (`Comparable<? super E> key = (Comparable<? super E>) x;`) appeared very strnage to be, since it appears to be a call to the comparable class as a generic, but there is the super inside the generic declaration, which is strange since super is normally reserved for calling the parent class's constructor.
The answer: Following this explanation on stackOverflow, it turns out that the super means that it is "some type that is an ancestor of E", which is notably different from <? extends E>, which is that it is "some type that is a descendent of E".

Source: https://stackoverflow.com/questions/1368166/what-is-a-difference-between-super-e-and-extends-e