

**CSE 373, Spring 2011
Midterm Key**

1. Big-Oh (17 Points)

```
SUM (i = 1 to n) { SUM (j = 1 to i / 2) { 4 } } - SUM (k = -50 to -1) { 1 }  
SUM (i = 1 to n) { 4 * SUM (j = 1 to i / 2) { 1 } } - 50  
SUM (i = 1 to n) { 4 * i / 2 } - 50  
2 * SUM (i = 1 to n) { i } - 50  
2 * ((n * (n + 1)) / 2) - 50  
n^2 + n - 50
```

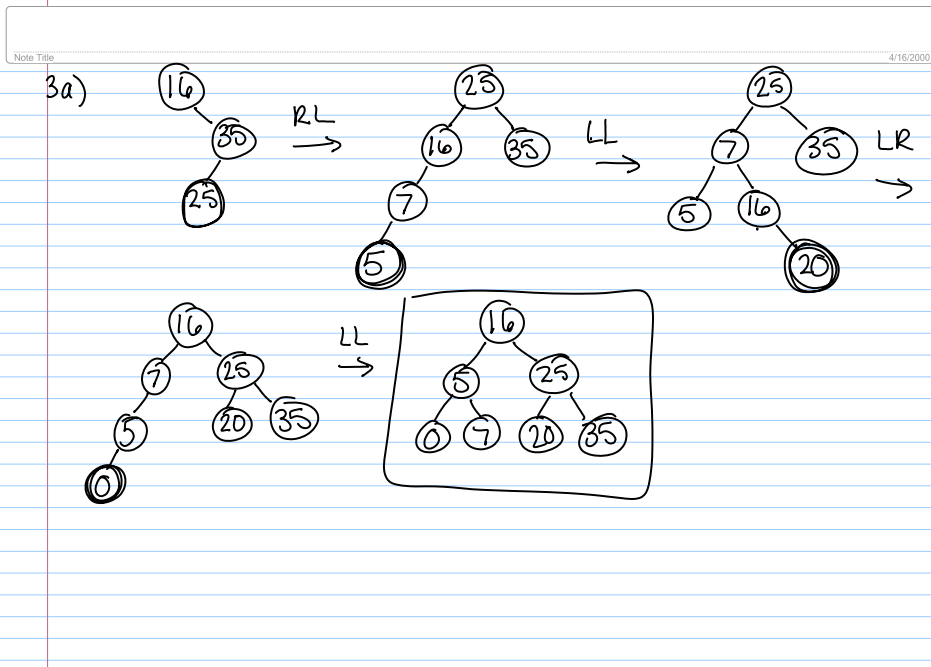
$O(n^2)$

2. Sorting (18 Points)

Part	Answer
a	insertion sort
b	merge sort
c	bubble sort
d	shell sort
e	quicksort
f	selection

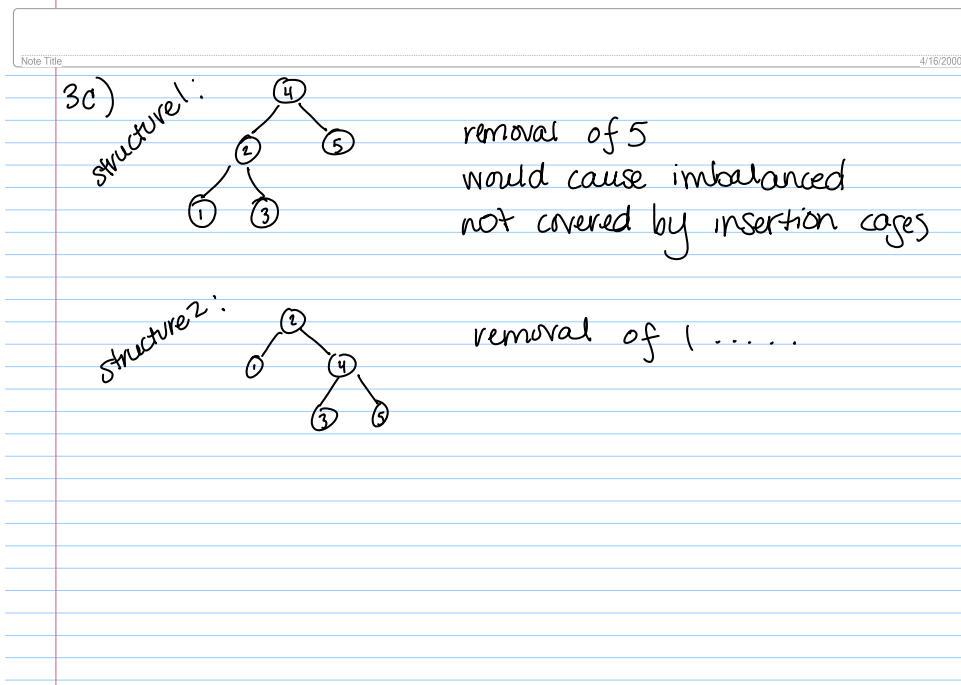
3. Trees (25 Points)

a.



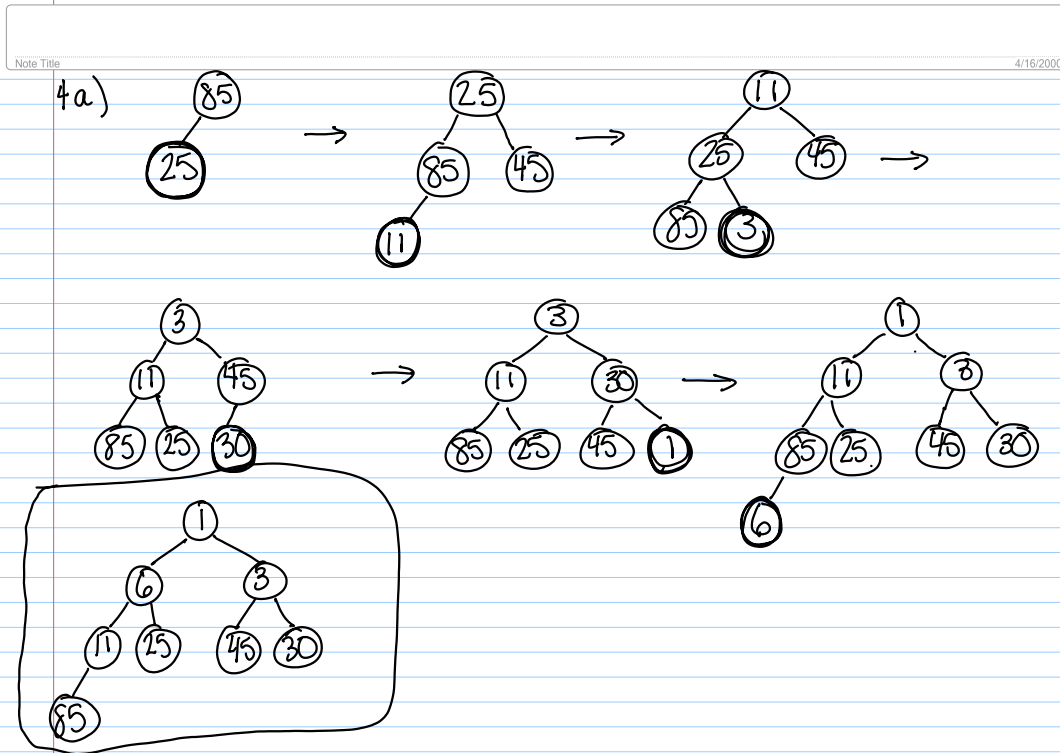
b. 0

c.

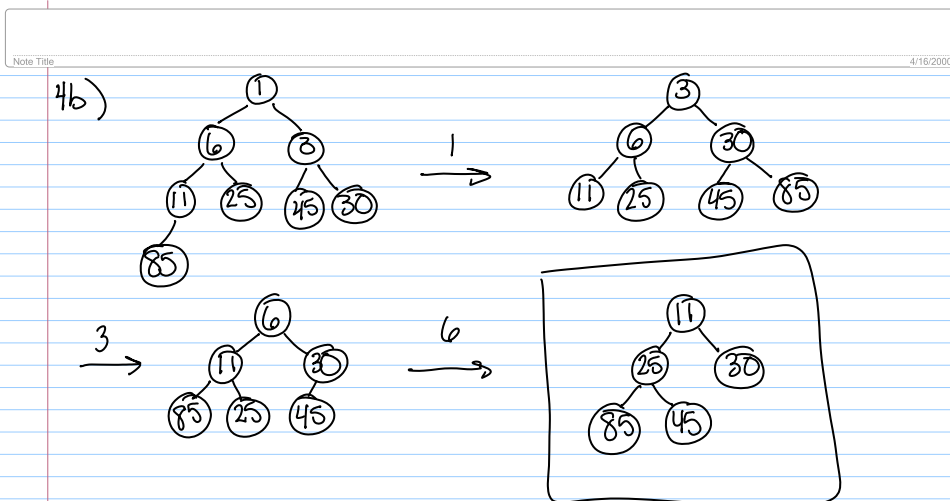


4. Heaps (20 Points)

a.



b.



5. Heap Implementation (20 Points)

```
public String nodesAtLevel(int level) {
    String str = "";
    int minIndexAtLevel, totalNodesAtLevel;
    minIndexAtLevel = totalNodesAtLevel = (int)Math.pow(2, level);

    if (level < 0 || size < minIndexAtLevel) {
        throw new IllegalArgumentException();
    }

    for (int i = 0; (i < totalNodesAtLevel) && (i + minIndexAtLevel <= size); i++) {
        str += array[i + minIndexAtLevel] + " ";
    }

    return str;
}
```