

CS 212 Homework #2

Please complete the homework problems on the following page. Note that this is an individual assignment and all work must be your own. Be sure to show your work when appropriate. This assignment is due by the beginning of class on Thursday, February 23, 2017. Note that there is no late turn-in meaning that this assignment must be turned in by the due date.

1. [2] Solve the recurrence relation $T(n) = 4T\left(\frac{n}{2}\right) + n^3$

2. [2] Solve the recurrence relation $T(n) = 3T\left(\frac{n}{2}\right) + n\log(n)$

3. [2] Solve the recurrence relation $T(n) = 3T\left(\frac{n}{3} + 5\right) + \frac{n}{2}$

4. [5] **Disjoint Sets.** Given the disjoint set $\{\{1\}, \{2\}, \{3\}, \{4\}, \{5\}, \{6\}, \{7\}, \{8\}, \{9\}, \{10\}\}$, perform the following operations using union-by-size and path compression when appropriate. Show the final result in both graph and array form.

Let $\text{Union}(x, y)$ be a function that performs a union on two disjoint sets. Assume that x and y do not belong to the same set.

Let $\text{Find}(x)$ be a function that returns the "root" of a given set. In this way, we can test whether or not two items belong to the same set.

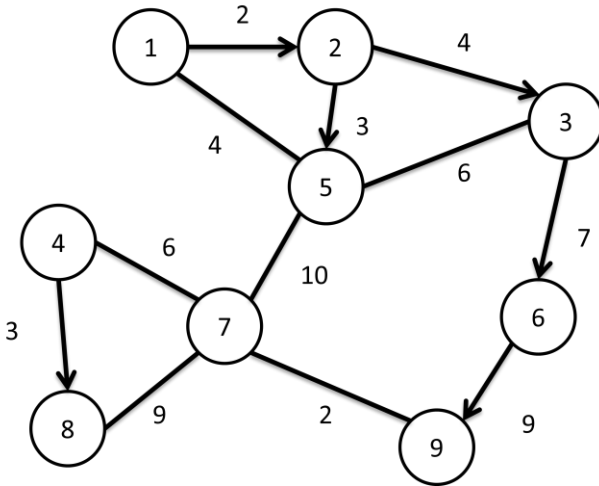
$\text{Union}(1, 2); \text{Union}(5, 6); \text{Union}(7, 10); \text{Union}(2, 9); \text{Union}(2, 10); \text{Union}(3, 5);$
 $\text{Union}(4, 3); \text{Union}(6, 8); \text{Find}(4) == \text{Find}(7); \text{Find}(9) == \text{Find}(10);$

Answer (Array):

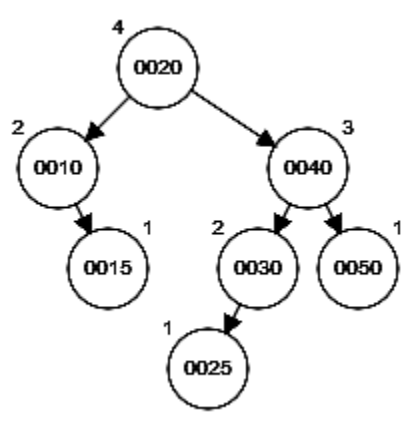
1	2	3	4	5	6	7	8	9	10

Answer (Graph):

5. [5] **Dijkstra's Algorithm.** Use Dijkstra's Algorithm to determine the shortest path starting at v1. Consider edges without arrow heads to be bidirectional. For example, v1 is connected to v5 and v5 is connected to v1 using the same edge. Use the table below to show your work.

[illegible]

7. [4] Insert 12 into the following AVL tree:

<p>Original state:</p>  <pre>graph TD; 0020((0020, 4)) --> 0010((0010, 2)); 0020 --> 0040((0040, 3)); 0010 --> 0015((0015, 1)); 0040 --> 0030((0030, 2)); 0040 --> 0050((0050, 1)); 0030 --> 0025((0025, 1));</pre> <p>The diagram shows an AVL tree with root 0020 (BF 4). Its left child is 0010 (BF 2), which has a right child 0015 (BF 1). Its right child is 0040 (BF 3), which has a left child 0030 (BF 2) and a right child 0050 (BF 1). Node 0030 has a left child 0025 (BF 1).</p>	<p>After Insert</p>
<p>After 1st Rotation (optional)</p>	<p>After 2nd Rotation (optional)</p>