Quiz/Homework Assignment #2
CSE326
Homework due date:
Wednesday, January 19th, beginning of class

January 14, 2000

This quiz is your second homework assignment. Any question you answer correctly in class will be considered correct for the purposes of the assignment. Any question you do not answer or which you answer incorrectly becomes a short answer question for the homework assignment. We will hand back the graded quiz on Friday.

Make sure that you write in your name and bubble in your UWID and section. Bubble your answers to the questions in the corresponding spaces on the answer sheet, and keep this question sheet so that you can work on any questions which you missed.

In italics are instructions on how to answer each question for homework. **A bold * precedes each correct answer.**

Good luck!

1. *Explain briefly little-o. Reduce each expression to simplest form. List most descriptive asymptotic relationship between each pair.*

   Which of the following are listed in strictly increasing asymptotic order (i.e., the left function is asymptotically less than, or little-o, of the right function)?

   (a) 1, 2
   (b) ***** $\log^2 n$, $n \log n + \log n$
   (c) n^2, $n\sqrt{n}$
   (d) $2^{\log n}$, $5n$
   (e) 3^n, n^3

2. *Derive bound for recurrence (as we did in class).*

   Which of the following is an asymptotically-tight (i.e. big-Θ) bound for the recurrence: $T(n) = T(n/2) + 1$?

   (a) $\Theta(n^2)$
   (b) $\Theta(n)$
   (c) ***** $\Theta(\log n)$
   (d) $\Theta(n \log n)$
   (e) $\Theta(1)$

3. *Note time for each statement. Write recurrence describing time taken by function. Derive bound for recurrence.*

   The following function is designed to make a computer user say that they see an albatross. It does this by asking them again and again until they go crazy and say that they do see one:

```
int Albatross(int n)
  If n=1 Then
    Return
  End If

  For i=1 to n
    Print ``Do you see the albatross now?''
  End For

  Albatross(n/2)
```

   Which of the following is an asymptotically-tight bound on the number of times the user would be asked about the albatross:

(a) $\Theta(n^2)$

(b) * $\Theta(n)$

(c) $\Theta(\log n)$

(d) $\Theta(n \log n)$

(e) $\Theta(1)$

4. *Derive correct equation (step by step).*

   Consider the following recurrence relation:

   $T(n) = 2T(n/3) + \sqrt{n}$

   Suppose we expand this equation k recursive times as we did in class (assume that the recurrence stated above is correct for $k = 1$). Then, in general we would have which of the following?

   (a) $T(n) = kn^{\log_3 2}$

   (b) * $T(n) = 2^k T(n/3^k) + \sum_{i=1}^{k} 2^{i-1}\sqrt{n/3^{i-1}}$

   (c) $T(n) = 2kT(n/3^k) + \sqrt{n/k}$

   (d) $T(n) = 2kT(n/3k) + \sqrt{n+1-k}$

   (e) $T(n) = k/2T(n/3k) + k\sqrt{k}$

5. *Justify correct answer (how do we know it?). Give correct form for (b), (d), and (e) (e.g., what is a good bound for the maximum number of leaves in the tree?). Briefly explain why (c) is not true.*

   A certain tree has a depth of d, a maximum branching factor of b, and n nodes total. Which of the following must be true of the tree?

   (a) * The tree has $n - 1$ edges.

   (b) The tree has no more than 2^d leaves.

   (c) The tree's smallest element is stored at its leftmost leaf.

   (d) n is no larger than d^b

   (e) Some node in the tree has d children.

6. *Name the nodes in each relationship (e.g., list nodes which are children of* **r***).*

   Which of the following is **not** equal to 3?

   (a) the number of children of **r**

   (b) the height of **p**

   (c) the number of ancestors of **t**

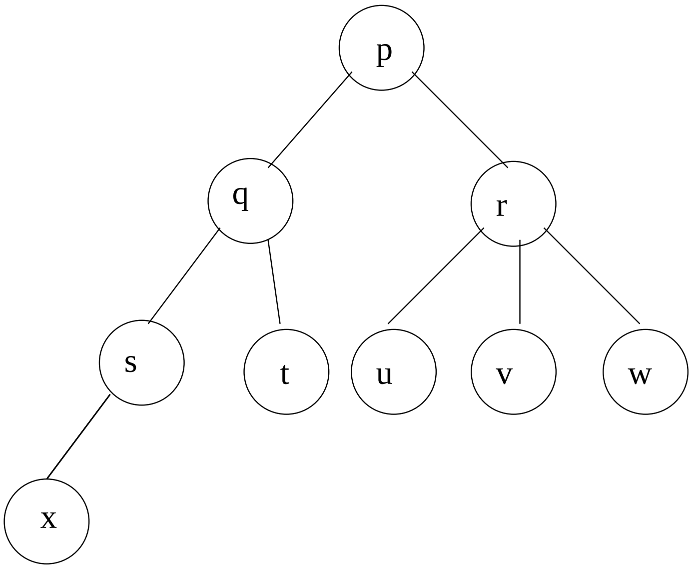

Figure 1: A tree with labelled nodes. Use this tree to answer questions 6-7.

   (d) the depth of **x**

   (e) * the depth of **u**

7. *Describe true relationship of each node in the answers (just one word answers are fine).*

   Which best describes the siblings of **u**?

   (a) **s**, **t**, **v**, **w**

   (b) * **v**, **w**

   (c) **p**, **r**

   (d) **u**, **x**

   (e) **s**, **t**, **u**, **v**, **w**

8. *Justify answer (why is it a good description), explain why (c) is a poor description, and explain how Priority Queues are similar to Stacks and Queues.*

   The Queue ADT can be briefly described as FIFO (First In First Out) and the Stack ADT as LIFO (Last In First Out). Which of these best describes the Priority Queue ADT?

   (a) Last In Last Out

   (b) * Always Best Out

   (c) Never Highest Out

   (d) Garbage In Garbage Out

(e) None of these; the Priority Queue ADT is totally unlike the Stack and Queue ADTs