Computer Science Foundation Exam

May 6, 2011

Section I A

COMPUTER SCIENCE

SOLUTION

**NO books, notes, or calculators may be used,
and you must work entirely on your own.**

Question #	Max Pts	Category	Passing	Score
1	10	DSN	7	
2	10	ANL	7	
3	10	ALG	7	
4	10	ALG	7	
5	10	ALG	7	
TOTAL	50			

You must do all 5 problems in this section of the exam.

Problems will be graded based on the completeness of the solution steps and not graded based on the answer alone. Credit cannot be given unless all work is shown and is readable. Be complete, yet concise, and above all be neat.

1) (10 points) **Recursion.** Write a RECURSIVE function, `printSideHouse(int min, int max)`, that prints rows of asterisks, starting with `min` stars on the first row, `min+1` stars on the second row, etc. After printing out `max` stars on a row, the following row should have `max-1` stars, with each subsequent row counting down until the last row has `min stars.` For example, if `printSideHouse(3, 5)` is called from main, the following will print to the screen:

```
***
****
*****
****
***
```

An auxiliary function printRow, is provided for your use. Please complete the empty function below so that it solves the given problem:

```
void printSideHouse(int min, int max)
{

     printRow('*', min);          // 2 pts

     if(min < max)                // 2 pts
          printSideHouse(min + 1, max); // 3 pts

     if(min != max)                // 1 pt
          printRow('*', min);     // 2 pts


    // Grading note: There are other ways to solve this.
    //               Just devote 3 pts to making sure the
    //               middle row prints once only.
}

int printRow(char ch, int n)
{
    int i;
    for (i=0; i<n; i++)
        printf("%c", ch);
    printf("\n");
}
```

2) (10 points) **Summations**

Determine a simplified closed-form solution for the following summation in terms of *n*:

$$\sum_{i=1}^{3n}\sum_{j=n+1}^{5n}(5i+3j)$$

$$\sum_{i=1}^{3n}\sum_{j=n+1}^{5n}5i+\sum_{i=1}^{3n}\sum_{j=n+1}^{5n}3j$$ **1 pt**

$$\sum_{i=1}^{3n}4n*5i+\sum_{i=1}^{3n}\sum_{j=n+1}^{5n}3j$$ **1 pt**

$$4n*5(3n(3n+1)/2)+\sum_{i=1}^{3n}\sum_{j=n+1}^{5n}3j$$ **2 pts**

$$90n^3+30n^2+\sum_{i=1}^{3n}(\sum_{j=1}^{5n}3j-\sum_{j=1}^{n}3j)$$ **1 pt**

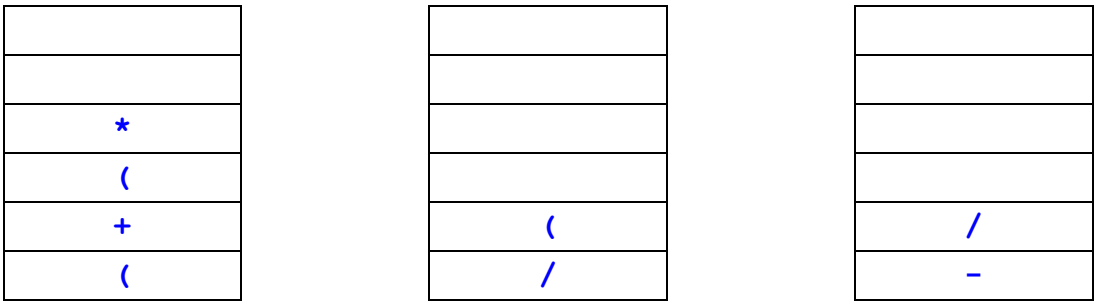
$$90n^3+30n^2+\sum_{i=1}^{3n}(\frac{3(5n)(5n+1)}{2}-\frac{3n(n+1)}{2})$$ **2 pts**

$$90n^3+30n^2+3n(36n^2+6n)$$ **2 pts**

$$198n^3+48n^2$$ **1 pt**

3) (10 pts) **Stack Applications.** Convert the following infix expression into its equivalent postfix expression using a stack. Additionally, you must show the contents of the stack at the indicated points (1, 2, and 3) in the infix expression.

```
                1                   2                   3
( A + ( B * C ) + D ) / ( E + F ) * G - H / I
```



Resulting Postfix expression:

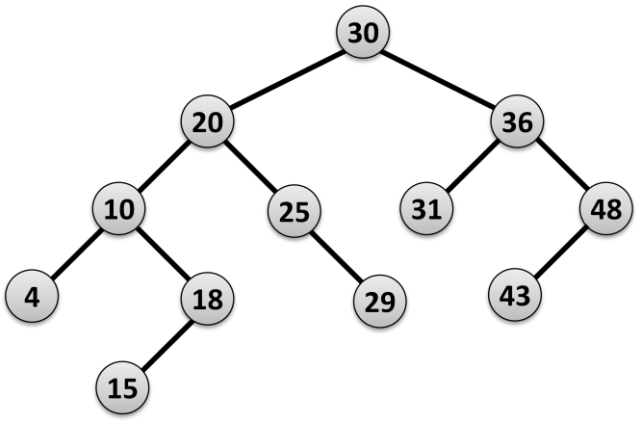
A	B	C	*	+	D	+	E	F	+	/	G	*	H	I	/	-

Grading: 2 pts for each stack, 4 pts for the expression total (take off 1 point per error)

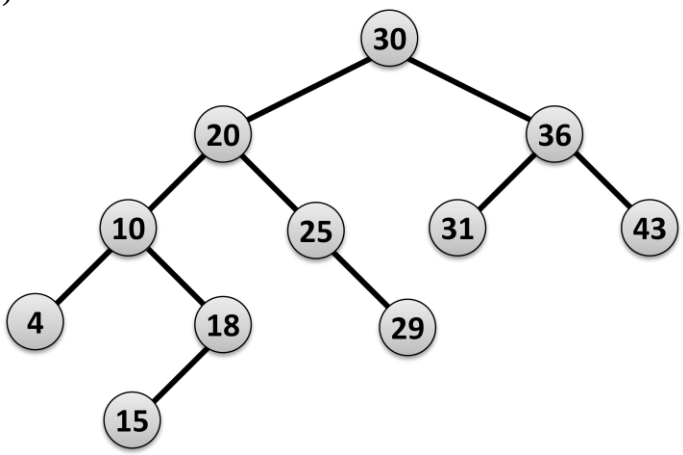
4) (10 points) **AVL Trees.** The tree shown below is a valid AVL tree.

(a) Show the state of the AVL tree immediately after deleting node 48 (the node that has 48 as a data value). Meaning, show the state before you "fix" (if necessary) the AVL tree.

(b) Show the state of the AVL tree after any necessary rebalancing.

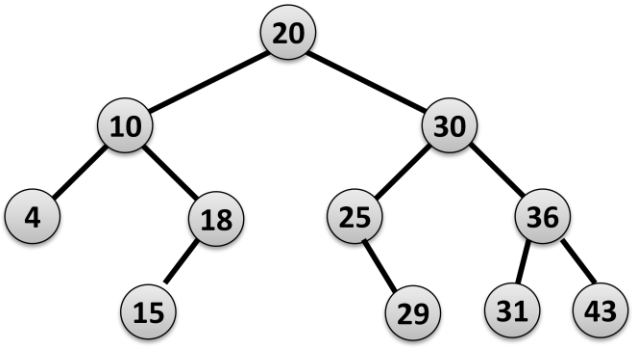


a)



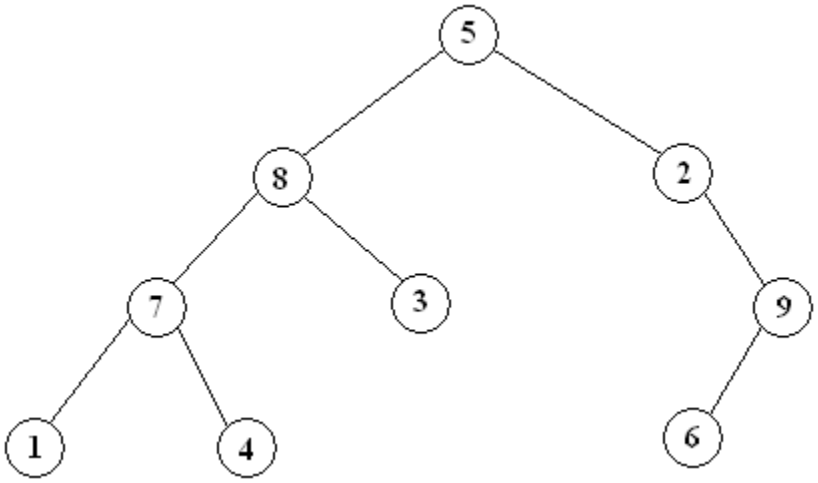
Grading: 3 points for this picture, just sliding the 43 into place.

b)



Grading: 1 pt for 20 at the root, 1 pt for 10, 1 pt for 30, 1 pt for each sub tree.

5) (10 points) **Binary Tree Traversals**



Give the preorder, inorder and postorder traversals of the binary tree shown above:

Preorder:

5, 8, 7, 1, 4, 3, 2, 9, 6 (3 points)

Inorder:

1, 7, 4, 8, 3, 5, 2, 6, 9 (4 points)

Postorder:

1, 4, 7, 3, 8, 6, 9, 2, 5 (3 points)

Grading note: If the traversals are correct but the names switched in any way, just take off 3 points. Decide partial credit based on the number of "changes" you have to make to their answer to get to the correct one by inserting one portion of the list into a different portion of the list.