Computer Science Foundation Exam

December 17, 2010

Section I A

COMPUTER SCIENCE

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

Name:____

PID: ____

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 <u>not</u> 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 <u>be neat</u>.

1) (10 points) Recursion. Write a recursive function that correctly prints the first n odd integers. You can assume that n > 0. For example, if your function takes in the number 5, your program should print 1 3 5 7 9 (the first 5 odd integers). Please make use of the function header below.

```
void printOddInts(int n)
{
     if (n == 1)
          printf("1 ");
     else {
          printOddInts(n-1);
          printf("%d ", 2*n-1);
}
```

Grading Criteria:
There are many ways to approach this problem. Be reasonable when grading.
Base case – 3 points (could also be n == 0…)
Recursive call – 4 pts, 2 for function, 2 for parameter
Print (last) – 3 pts, 1 for printf, 1 for % code, 1 for value printed

2) (10 points) **Summations**

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

$$\sum_{i=n+1}^{4n}\left(2i+\sum_{j=1}^{n}4j\right)$$

b) Evaluate the following summation: $\sum_{j=1}^{10}\left(\sum_{i=21}^{40}2i\right)$

Correct Answers:

a)

$$\sum_{i=n+1}^{4n}\left(2i+\sum_{j=1}^{n}4j\right)=\sum_{i=n+1}^{4n}\left(2i+\frac{4n(n+1)}{2}\right)=\sum_{i=n+1}^{4n}\left(2i+2n(n+1)\right)=\sum_{i=n+1}^{4n}2i+\sum_{i=n+1}^{4n}2n(n+1)$$

$$=\sum_{i=1}^{4n}2i-\sum_{i=1}^{n}2i+\sum_{i=1}^{4n}2n(n+1)-\sum_{i=1}^{n}2n(n+1)=2\frac{4n(4n+1)}{2}-2\frac{n(n+1)}{2}+8n^2(n+1)-2n^2(n+1)$$

$$=4n(4n+1)-n(n+1)+6n^2(n+1)=16n^2+4n-n^2-n+6n^3+6n^2=6n^3+21n^2+3n$$

b)

$$\sum_{j=1}^{10}\sum_{i=21}^{40}2i=\sum_{j=1}^{10}2\left(\sum_{i=1}^{40}i-\sum_{i=1}^{20}i\right)=\sum_{j=1}^{10}2\left(\frac{40(41)}{2}-\frac{20(21)}{2}\right)=\sum_{j=1}^{10}2\big(20(41)-10(21)\big)$$

$$=\sum_{j=1}^{10}2\big(820-210\big)=\sum_{j=1}^{10}2\big(610\big)=\sum_{j=1}^{10}1220=1220\sum_{j=1}^{10}1=1220(10)=12{,}200$$

Grading Criteria:

a)

Properly dealing with the inner summation – 1 point
Properly splitting the summation into two parts – 1 point
Properly dealing with limits of the summation – 2 points
Simplifying the resulting closed form – 1 point

b)

Properly dealing with the inner summation – 4 points
- Properly splitting the inner summation into two parts – 2 points
- Evaluating the two resulting summations properly – 2 points

Properly dealing with the outer summation and determining the final answer – 1 point

---

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  )
```

1	2	3
+		(
(	(	*
*	–	–

Resulting Postfix expression: **A B C + * D / E F + G H - * -**

Grading Criteria:
Stack 1 – 3 points
Stack 2 – 2 points
Stack 3 – 3 points
Final answer – 2 points

4) (10 points) **Hash Tables.** Insert the following numbers (in the order that they are shown…..from left to right) into a hash table with an array of size 10, using the hash function, H(*x*) = *x* mod 10.

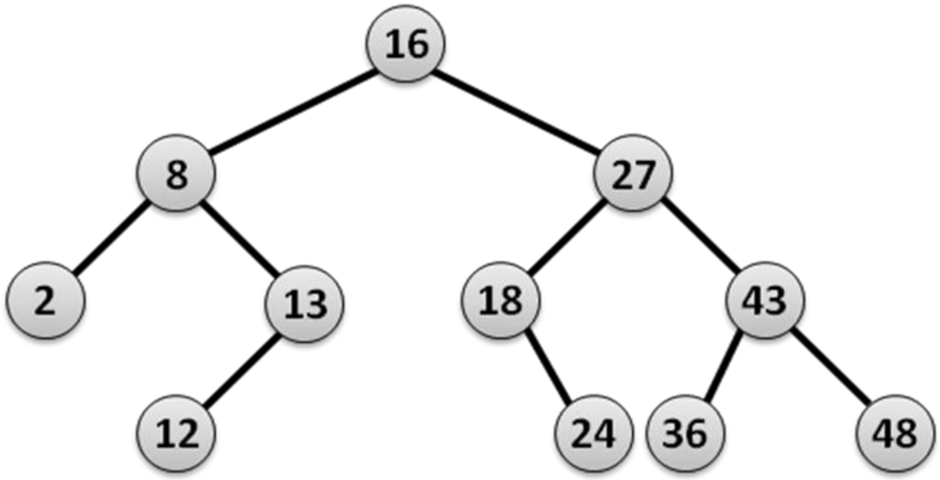
277, 522, 312, 188, 527, 437, 248, 219

Show the result of the insertions when hash collisions are resolved through **a)** linear probing, **b)** quadratic probing, and **c)** separate chaining (where each item is added to the BACK of the appropriate linked list).

	a)		**b)**		**c)**	
0	**437**	0	**219**	0		
1	**248**	1	**527**	1		
2	**522**	2	**522**	2	**522**	**-> 312**
3	**312**	3	**312**	3		
4	**219**	4		4		
5		5		5		
6		6	**437**	6		
7	**277**	7	**277**	7	**277**	**-> 527 -> 437**
8	**188**	8	**188**	8	**188**	**-> 248**
9	**527**	9	**248**	9	**219**	

Grading Criteria:
3 points for linear probing (a) (1 pt off if 1 or 2 wrong, 2 pts off if at least 2 correct)
4 points for quadratic probing (b) (1/2 pt for each, round down)
3 points for separate chaining (c) (1 pt off if 1 or 2 wrong, 2 pts off if at least 2 correct)

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



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

Correct Answers:

Preorder:

16, 8, 2, 13, 12, 27, 18, 24, 43, 36, 48

Inorder:

2, 8, 12, 13, 16, 18, 24, 27, 36, 43, 48

Postorder:

2, 12, 13, 8, 24, 18, 36, 48, 43, 27, 16

<u>Is the tree depicted above an AVL tree? State Yes or No and briefly explain.</u>

Yes, because the balance factor at every node is within an acceptable range (-1, 0, or 1).

Grading Criteria:

3 points for each traversal (2 pts if mostly correct, 1 pt if a few correct)

1 point for answering the AVL tree question.