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: __________________________________________

<table>
<thead>
<tr>
<th>Question #</th>
<th>Max Pts</th>
<th>Category</th>
<th>Passing</th>
<th>Score</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10</td>
<td>DSN</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>10</td>
<td>ANL</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>10</td>
<td>ALG</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>10</td>
<td>ALG</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>10</td>
<td>ALG</td>
<td>7</td>
<td></td>
</tr>
<tr>
<td>TOTAL</td>
<td>50</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

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 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.

```c
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)  
\[
\sum_{i=n+1}^{4n} \left( 2i + \sum_{j=1}^{n} 4j \right) = \sum_{i=n+1}^{4n} \left( 2i + \frac{4n(n+1)}{2} \right) = \sum_{i=n+1}^{4n} (2i + 2n(n+1)) = \sum_{i=n+1}^{4n} 2i + \sum_{i=n+1}^{4n} 2n(n+1)
\]

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

\[
= 4n(4n+1) - n(n+1) + 6n^2(n+1) = 16n^2 + 4n - n^2 - n + 6n^3 + 6n^3 = 6n^3 + 21n^2 + 3n
\]

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

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

**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.

\[
\begin{align*}
\text{A} & \times ( \text{B} + \text{C} ) / \text{D} - ( \text{E} + \text{F} ) \times ( \text{G} - \text{H} ) \\
\end{align*}
\]

Resulting Postfix expression: \( \text{A B C + * D / E F + G H - * -} \)

**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).

<table>
<thead>
<tr>
<th></th>
<th>a)</th>
<th></th>
<th>b)</th>
<th></th>
<th>c)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>437</td>
<td>0</td>
<td>219</td>
<td></td>
<td>0</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>248</td>
<td>1</td>
<td>527</td>
<td>1</td>
<td>2</td>
<td>522</td>
</tr>
<tr>
<td>2</td>
<td>522</td>
<td>2</td>
<td>522</td>
<td>2</td>
<td>522</td>
<td>312</td>
</tr>
<tr>
<td>3</td>
<td>312</td>
<td>3</td>
<td>312</td>
<td>3</td>
<td>522</td>
<td>312</td>
</tr>
<tr>
<td>4</td>
<td>219</td>
<td>4</td>
<td></td>
<td>4</td>
<td>219</td>
<td>312</td>
</tr>
<tr>
<td>5</td>
<td></td>
<td>5</td>
<td></td>
<td>5</td>
<td></td>
<td>312</td>
</tr>
<tr>
<td>6</td>
<td></td>
<td>6</td>
<td>437</td>
<td>6</td>
<td></td>
<td>312</td>
</tr>
<tr>
<td>7</td>
<td>277</td>
<td>7</td>
<td>277</td>
<td>7</td>
<td>277</td>
<td>312</td>
</tr>
<tr>
<td>8</td>
<td>188</td>
<td>8</td>
<td>188</td>
<td>8</td>
<td>188</td>
<td>312</td>
</tr>
<tr>
<td>9</td>
<td>527</td>
<td>9</td>
<td>248</td>
<td>9</td>
<td>219</td>
<td>312</td>
</tr>
</tbody>
</table>

**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

**Is the tree depicted above an AVL tree? State Yes or No and briefly explain.**

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.