Data Structures

Midterm Exam, Fall 2015

01. **(30%)** Explain the following terms and terms comparisons:

(a) Tree traversal

(b) LIFO lists vs. FIFO lists

(c) The level of a tree node

(d) Row major order

(e) Stack underflow

(f) The degree of a tree

(g) Full binary tree

(h) Memory leakage problem

(i) Algorithms vs. programs

(j) Performance analysis vs. performance measurement

02. **(7%)** Answer “True” or “False” for the following statements.

(a) An empty binary tree is invalid while a tree may have zero nodes.

(b) The order of children is irrelevant in a binary tree.

(c) The order of operands is the same in infix and postfix representation from for all math expressions.

(d) A binary tree us complete binary if all the leaf nodes are on two adjacent levels.

(e) An n-node ($n > 0$) complete binary tree is not unique.

(f) A k-level ($k \geq 1$) full binary tree is unique.

(g) Level order binary tree traversal requires a stack while preorder traversal needs a queue to realize the traversal operation.

03. **(8%)** Prove or disprove the following statements:

(a) $\sum_{i=0}^{n} i^3 = \Theta(n^4)$

(b) $100n^2 + 200 = O(n)$

(c) $n! = O(n^n)$

(d) $n^{1.001} + n\log n = \Theta(n^{1.001})$

04. The system stack is a typical application or stacks. Please answer the following questions relevant to system stacks.

(a) **(4%)** What fields are required in an AR?

(b) **(5%)** Please explain the life cycle of an AR (When is an AR created? When is it deleted? What is the detailed processing during the lifetime of an AR? Etc.)

05. **(a)** **(2%)** What is the equivalence determination problem?

(b) **(6%)** Please explain how to solve the equivalence determination problem. Your answer should include necessary pseudo code. Note that it is not necessary to follow the design introduced in the textbook and creative ideas are encouraged.

(c) **(2%)** Let m and n represent the number of related pairs and the number of objects, respectively. Determine the overall time complexity of your pseudo code in (b).

06. **(9%)** Assume that it takes two units of memory location to store an integer and row major order is adopted. Consider the following array declaration:

```
int A[5][6][8];
```

(a) If `A[0][0][0]` is stored at address 2000, calculate the memory address of `A[2][0][7]`.

(b) If `A[0][0][0]` is stored at address 2000, indicate which array element is at the location 2060.

(c) If `A[3][0][0]` is stored at address 2000, calculate the memory address of `A[1][5][5]`.

07. **(a)** **(2%)** During the process of transforming a parenthesized infix expression to a postfix one, why do we need two types of precedence, an *in-stack precedence* and an *incoming precedence*?

(b) **(2%)** Write the postfix form of the following expressions:

(i) $A-B*D+E/F+A*D+C$

(ii) $(A-B)*D+E/(F+A*D)+C$

08. **(6%)** Please explain how to perform “in place” chain inversion of a singly linked list. (It is not necessary to follow the design introduced in the textbook.)

09. **(5%)** Explain how to implement a circular queue by using an array and why a circular queue of size MAX_QUEUE_SIZE can be hold at most MAX_QUEUE_SIZE-1 elements.

10. **(3%)** Given an in-order sequence BAECDJHFGI and a post-order sequence ABCDEFGHIJ, can you derive a unique binary tree? If yes, draw the binary tree; or you have to give two distinct binary trees which can generate above sequences.

11. **(5%)** For any nonempty binary tree, T, if n_0 is the number of leaf nodes and n_2 is the number of nodes of degree 2, prove that $n_0 = n_2 + 1$.

12. **(8%)** Explain how a compiler performs evaluation of expressions without parentheses, especially the role of stacks in the processing procedure.

13. **(a)** **(8%)** What is the left child-right sibling representation? Explain pros and cons of this representation as compared with the linked list representation.

(b) **(8%)** Explain how to implement a binary tree by using an array. Note that you have to point out the relationship of indices between a node *i* and its parent, left child, and right child. Also, explain pros and cons.