CSE 5350/7350 **Due Date:**
D.W. Matula **10 Oct 2013**

Homework Set # 2

Data Structure Engineering for Algorithm Design

(70 points + 10 bonus points)

Data Structures deals with the **representation** and **organization** of data to support efficient:

- **storage**
- **access**
- **execution of operations**
- **display of results**.

---

1. **[Data structure conversion](15 points)**. Design and analyze an on-the-fly algorithm for converting a binary search tree to an ordered doubly linked list following the inorder traversal of the data. Utilize a loop invariant (text p.17) to argue the correctness of your algorithm. Have your algorithm use as little extra space as possible, with "in place" the goal. Perform a walkthrough for the sequence S where the elements of S are first processed left-to-right to build the search tree. Then apply your algorithm to convert the binary search tree to a doubly linked list, showing the status of the conversion at each node of the search as it is consumed and moved into the doubly linked list.

   S = 63, 35, 97, 49, 83, 28, 20, 45, 70, 12, 54, 78

2. **[Heapify] (15 points).** The array form of a heap (i.e. a binary heap array) has the children of the element at position i in positions (left @ $2i$, right @ $2i+1$). For a balanced ternary heap array the three children of the element at position i are at positions (left @ $3i-1$, middle @ $3i$, right @ $3i+1$). The term "Heapify" refers to the algorithm that builds a binary heap array from right-to-left (leaf-to-root in binary tree form) given the size of the array to be built. The term "Ternary Heapify" here refers to the analogous algorithm that builds a balanced ternary heap array in right-to-left order.

   a. Heapify the sequence S of Problem 1, and give the number of comparisons required.

   b. Do the same as (a) for Ternary Heapify.

   c. Build the decision tree determined by the comparisons employed to Heapify a six element list: a_1, a_2, a_3, a_4, a_5, a_6.What is then the worst case and average case number of comparisons to Heapify six elements? Is this optimal in either case?

   d. Do the same as (c) for Ternary Heapify.

3. **[Graph (Symmetric Matrix) Reordering]** Let the adjacency list representation of a graph (0, 1 symmetric matrix) be given by a packed edge list. Describe an algorithm for reordering the edge list in place so all adjacency lists follow a new vertex ordering given by a permutation p(1), p(2), ..., p(n). Apply your algorithm to the graph given by the packed edge list stored as example HW2 - 3 graph. Reorder by determining a smallest last search order with "ties" broken lexicographically.

HW2 - 3 graph in adjacency list form:

- a: c, f, g, h;
- b: c, d;
- c: a, b, g, h;
- d: b, e, f, g;
- e: d, f, g;
- f: a, d, e, h;
- g: a, c, d, e
- h: a, c, f;

4. **[Sparse Matrix Multiplication]** Describe an efficient sparse matrix multiplication procedure for two n x n matrices stored in "adjacency list" form. Assume matrix A has m_A non zero entries and matrix B has m_B non zero entries, with $n << m_A << m_B << n^2$. Describe why your algorithm will generate the n × n matrix product in adjacency list form in time O (n m_A).

Apply your algorithm for multiplying the two 8 × 8 0,1 matrices A_1, B_1 where A_1 is given by the following adjacency list of unit entries and B_1 is given by the adjacency list of unit entries of Problem 3. Count the actual number of multiplications used in this matrix product and explain why it is considerably less than the bound n m_A (which is 8 × 19 =152) in this case.

Adjacency List For 0,1 Matrix A_1

- 1: 3, 5, 7
- 2: 4, 8
- 3: 5, 8
- 4: 2, 6
- 5: 1, 2, 7, 8
- 6: 3
- 7: 1, 5, 6
- 8: 6, 7

5. **[Random Geometric Graphs: Generation and Display]**

i. Generate and display a random geometric graph (RGG) with n = 200 vertices and r = 0.16.

ii. Generate an RGG with n = 1600 vertices and r = 0.06. Display all the vertices and only the edges incident to either a particular vertex of the highest degree or a particular vertex of lowest degree (give the degrees of these two vertices).

iii. [5 bonus points] Generate an RGG with 400 vertices on the unit sphere with r = 0.25, where distance is measured between the (x, y, z) coordinates. Then display the graph for the hemisphere with $z \geq 0$, and separately for the hemisphere with $z \leq 0$.

6. **[Floating Point Number Formats]**

a) What is the probability that a random 64-bit word represents a:

i. finite binary floating point number, and

ii. a normalized binary floating point number, and,

b) Do the same as (a) for a random 128-bit word.

c) [5 bonus points] Do the same as (a) and (b) for decimal floating point numbers in the 64 and 128 bit formats.