CSE 373 Winter 2003

Data Structures and Algorithms

Assignment #4

Due: Friday February 14

You are strongly encouraged to look at this assignment before the midterm, in particular the very short exercises 1 and 4.

1. (8 points) Weiss Problem 5.1

2. (4 points) Weiss Problem 5.2. Do the rehash only for parts (b) and (c) of Problem 5.1.

3. (12 points)

In this problem you will design algorithms to handle insertion, deletion, and find for open addressing hashing (aka closed hashing) using linear probing.

Assume that we want to insert non-negative integers in a table of size M using the hashing function *hash* (*hash*(*i*) returns an integer between 0 and M-1; you don't have to write the *hash* function). Initially the table is empty and all entries are set to -2. When an entry has been inserted and then deleted, the entry becomes -1.

Give pseudocode algorithms for the following 3 functions

- Find(x,T) which returns the index of where x is stored if it is in the table T and -1 otherwise.
- Insert(x,T) which inserts x into the table T. You can assume that x is not currently in the table. Naturally, a location with a -1 or a -2 in it can be used to place a newly inserted item. Your algorithm should indicate when the hash table T is full and x cannot be inserted (for example, a successful insertion can return a value of 0 and an unsuccessful one a value of 1).
- Delete(x,T) which removes x from the table T replacing x by the value -1 in the location where x was. You can assume that x is in the table.

You are not asked to implement a Rehash function.

4. (6 points) Weiss Problem 6.8

5. (10 points)

A 3-heap is like a binary heap except that each node has 3 children instead of 2 (cf. Figure 6.19; Of course one of the nodes at the next to last level might have only 1 or 2 children).

Assuming a 3-heap is stored in an array indexed 1 to Max, give formulas for computing the index of the 3 children of the node at index i. Give a formula for computing the parent of a node at index i. Using pseudocode, design:

- PercolateDown(i : integer, x : keyvalue) for the 3-heap.
- PercolateUp(i : integer, x: keyvalue) for the 3 heap.