CSE 326 Problem Set 4 November 10, 1997

This problem set is not to be handed in, but solving the problems will help you prepare for the midterm.

1. Recall the operations of split and join for splay trees. Split takes a splay tree T and key x and returns two splay trees T_1 and T_2 such that T_1 contains all the keys in T that are $\leq x$ and T_2 contains all the keys in T that are $> x$. Join operates on two splay trees T_1 and T_2 where all the keys in T_1 are less than all the keys in T_2. It returns a splay tree T which contains all the keys in either T_1 or T_2.

   Describe an algorithm that uses split and join operations to do range deletion. Range deletion takes a splay tree T together with keys x and y and returns a splay tree T' which contains all the keys z in T except those where $x \leq z \leq y$. In other words, range deletion removes all the keys between x and y, inclusive.

   You can use an abstract drawing or pseudocode to describe your algorithm. Your algorithm should use just a few split and join operations to do the job of range deletion. Explain why your algorithm is correct.

2. What is the minimum number of keys that can be stored in a B-tree of order m and height h? A tree with a single node is of height 0.

3. Double hashing is a closed hashing method that requires two hash functions $h_0(x)$ and $k(x)$. The sequence of addresses probed when inserting key x into the hash table is $h_0(x), (h_0(x) + k(x)) \bmod M, (h_0(x) + 2k(x)) \bmod M, \cdots$. Show the steps in inserting the following sequence of keys when $M = 7$, $h_0(x) = x \bmod 7$ and $k(x) = (x \bmod 6) + 1$: $73, 19, 56, 45, 84, 23, 43$.

4. A spell checker reads an input file and prints out all the words not contained in an online dictionary. A simple strategy for building a spell checker is to insert each word of the dictionary into a hash table. Then, whenever a fill is to be spell-checked, a find operation on the hash table is performed on each word of the input file.

   An alternative method, which is not foolproof but will catch most spelling errors, is to choose a hash function h which maps the universe of all possible words to the range $\{0, 1, \cdots, M-1\}$, and then construct an array H of bits, where, for $i = 0, 1, \cdots, M-1$, $H[i]$ is equal to i if there exists a word x in the dictionary such that $h(x) = i$. Then, whenever a file is to be spell checked, the spell checker computes hash address $h(y)$ of each word in the input file. If $H[h(y)] = 0$ then it declares that y is not in the dictionary; if $H[h(y)] = 1$ then it declares that y is in the dictionary.

   (a) How would you construct the array H?

(b) Does this algorithm ever falsely declare that a word is in the dictionary?

(c) Does this algorithm ever falsely declare that a word is not in the dictionary?

(d) Assume that, whenever a word is hashed into H for the first time, its hash address is equally likely to be any address in H. How frequently will the algorithm make an error if there are $30,000$ words in the dictionary and $M = 300,007$?

(e) Consider extendible hashing with $M = 4$. Indicate the contents of the directory and of each page after the insertion of keys with the following sequence of hash addresses:
$00110, 10101, 01100, 01101, 11110, 00011, 11111, 01011$.

5. (a) Show the result of inserting $10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13$ and 2, one at a time, into an initially empty binary heap.

   (b) Show the steps in applying Heapsort to the above keys, assuming they are initially stored in an array in the order of their occurrence above.

6. Weiss, problem 6.15

7. If a d-heap is stored as an array, what are the addresses of the parent and of the children of the node with address i?

8. (a) Merge the two heaps shown in Weiss, Figure 6.57, using the merge algorithm for leftist heaps.

   (b) Merge the same two heaps, using the merge algorithm for skew heaps.