Homework 4

Fundamental Algorithms, Fall 2002, Professor Yap

Due: Thursday Nov 14, in class

INSTRUCTIONS:

- This homework is related to Professor Siegel's lecture as well as our lecture on dynamic programming.
- TIP OF THE DAY: It is highly recommended that you rewrite your solution neatly for submission, this time paying attention to missing assumptions, unclear notations, etc. You might be surprised at why you discover.

1. **Merging of Two Sorted Lists, (10+10 Points)**

   Solve parts (a) and (b) of problem 8.6 in page 181 of the CLR textbook.

   REMARKS: You may find Stirling's approximation (p.55) for the factorial function useful. Note that to merge a sorted list of m keys with a sorted list of n items, the standard algorithm takes $m+n-1$ comparisons. Parts (c) and (d) are not graded, but we ask you to attempt it (we will sketch the solution).

2. **LCS and Edit Distances (10 Points each part)**

   Let $X = agacgttcgtta$ and $Y = cgactgctgta$.

   - **(i)** Compute $L(X, Y)$, which is the length of any longest common subsequence of X, Y. Be sure to show a matrix containing your computation.
   - **(ii)** Using the matrix you made in part (i), draw arrows from each square to indicate where each entry was derived from. Next, use these arrows to come up with a longest common subsequence of X and Y.
   - **(iii)** Compute the edit distance $D(X, Y)$, which was defined in recitation to be the minimum number of Insert/Delete/Replace steps to transform X to Y. Again, please show a matrix with your computation.
   - **(iv)** Using the matrix you made in part (iii), draw arrows on your grid to indicate where each entry was derived from. Next, using these arrows to come up with a sequence of $D(X, Y)$ Insert/Delete/Replace instructions to transform X into Y.
   - **(v)** Let X and Y be strings and A the associated matrix containing the computation record for $L(X, Y)$ (as in part(i)). Describe an algorithm in psuedocode which, given (X, Y, A), computes a longest common subsequence of X and Y.

- **(vi)** Solve the analogous problem to part (v), but this time for the edit distance problem $D(X, Y)$.

3. **Generalized LCS (15+30+10 Points)**

   Recall the recurrence relation we showed for $L(X, Y)$.

   (a) Derive and prove the analogous recurrence for $L(X, Y, Z)$ is the length of any longest common subsequence of three strings X, Y, Z.

   (b) Describe how you would organize the computation of $L(X, Y, Z)$ in a systematic way.

   (c) Illustrate your solution to (b) by computing $L(X, Y, Z)$ where $X = longest$, $Y = lengthen$ and $Z = elongated$.

4. **Matrix Chain Product (15 Points)**

   Solve the Matrix Chain Product problem for a matrix chain with 8 matrices, with dimension:

   $$(2, 4, 3, 3, 7, 1, 9, 8, 5)$$

   Please organize your work in a half-matrix as illustrated in Figure 15.3 (p.337, CLR).

   When you are done, please draw the actual parenthesis structure for evaluating this chain.

5. **Treaps** $(10 + 50$ **Points)**

   A treap is a binary search tree combined with a heap structure. Read the description of treaps on pages 296-297 in the CLR textbook.

   - **(i)** Construct a Treap out of the following (key, priority) pairs: $(12, 2)(15, 6)(5, 1)(7, 3)(2, 4)(17, 5)$
   - **(ii)** Do parts (a), (b), (c), and (d) of problem 13-4 on pages 296-298 in the CLR textbook.

     Hints:

     – For part (b), it might help to look at section 12.4, pages 265 - 268 in the CLR book. Alternatively, you can try the following: Let $H(n)$ denote the height of a binary tree with n nodes. This is a random variable (so you can take expectation, etc). If i is the number of children in the left subtree of the root, then: $H(n) = 1 + max\{H(i), H(n-1-i)\}$. Hence,

     $$E[H(n)] = 1 + \frac{1}{n}\Sigma_{i=0}^{n-1} max\{E[H(i)], E[H(n-1-i)]\}.$$

     Then use real induction to prove that for some constants $0 < c_1 \leq c_2$ that $c_1 lg(n) \leq E[H(n)] \leq c_2 lg(n)$.

     – For part (c), first give the English description, followed by pseudocode. If necessary, add comments to your pseudocode.