Data Structure and Algorithm II

Homework #1

Due: 5pm, Thursday, March 17, 2011

=== Homework submission instructions ===

- Submit the answers for writing problems (including your programming report) through the CEIBA system (electronic copy) or to the TA in R432 (hard copy). Please write down your name and school ID in the header of your documents. You also need to submit your programming assignment (problem 1) to the Judgegirl System(`http://katrina.csie.ntu.edu.tw/judgegirl/`).

- Each student may only choose to submit the homework in one way; either all as hard copies or all through CEIBA except the programming assignment. If you submit your homework partially in one way and partially in the other way, you might only get the score of the part submitted as hard copies or the part submitted through CEIBA (the part that the TA chooses).

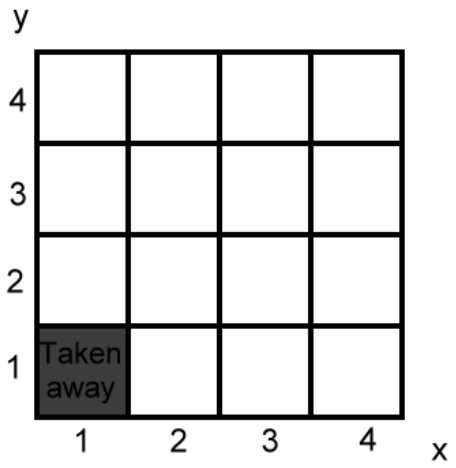
- If you choose to submit the answers of the writing problems through CEIBA, please combine the answers of all writing problems into only one file in the doc/docx or pdf format, with the file name in the format of "hw1_[student ID].{pdf,docx,doc}" (e.g. "hw1_b98902010.pdf"); otherwise, you might only get the score of one of the files (the one that the TA chooses).

Problem 1. (30%) Assume that n is a positive integer which is a power of two. We have a chessboard which is of size n x n with a randomly selected single tile taken away (see Figure 1(a)). You are asked to cover the entire chessboard with the piece shown in Figure 1(b). Note that the piece can be rotated. All the tiles on the chessboard except the one taken way have to be covered, and no tiles can be left outside of the chessboard. In this problem, we ask you to design a program to solve this problem with the divide-and-conquer strategy. In this problem, 10 points for report and 20 points for 10 test cases.

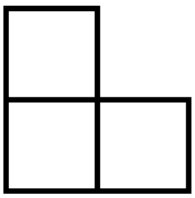
Input: 2 lines. n, the size of the chessboard, $2 \leq n \leq 2048$, is in the first line. x y, the coordinate of the tile which is taken away, $1 \leq x, y \leq n$, is in the second line.
Output: $(n^2 - 1)/3$ lines. In each line, we output a tuple, x y d. Note that the lines are sorted by x and y in the ascending order. x and y represents the coordinate of the lower left tile of the piece. d is one of the following: {UL, UR, LL, LR}, representing the missing tile of the piece in Upper Left, Upper Right, Lower Left, or Lower Right. You are permitted to use qsort in C standard library to sort your output.
Example (Figure 1(c)):
Input:
4
1 1
Output:
1 1 LL
1 3 LR
2 2 LL
3 1 UL
3 3 LL
Please also submit a report in which you give a clear description of your algorithm. Analyze the running time of your algorithm by using the recurrence.

Problem 2. (20%) Given a set of n points in a 3-dimensional space, (x, y, z), derive an algorithm based on the divide-and-conquer strategy to determine the k closest pairs of points. n and k are positive integers and $k < n$. Write down the recurrence to represent the running time of your algorithm and solve the recurrence. Note that there are many ways to solve this problem; please come up with your own.

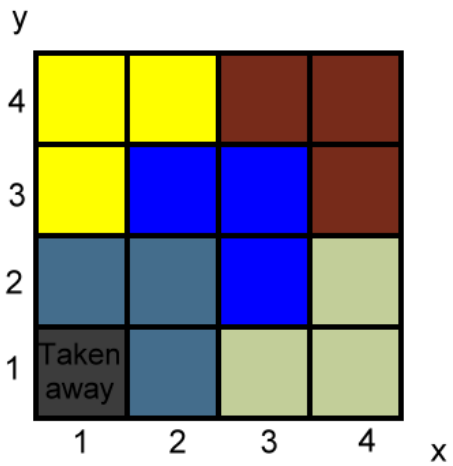
Problem 3. (20%) Consider the multiselection problem: given a set S of n elements and a set K of r ranks $k_1, k_2, ..., k_r$, find the k_1th, k_2th, ..., k_rth smallest elements. For example, if $K = \{2, 7, 9, 50\}$, the problem is to find the 2nd, 7th, 9th, and 50th smallest elements. This problem can be solved trivially in $\Theta(rn)$ by using the selection algorithm we talked about in the class (run the algorithm for r times, once for each rank k_j, $1 \leq j \leq r$. Give an $O(n \log r)$ time algorithm to solve this problem.



(a) The chessboard with 1 tile taken away

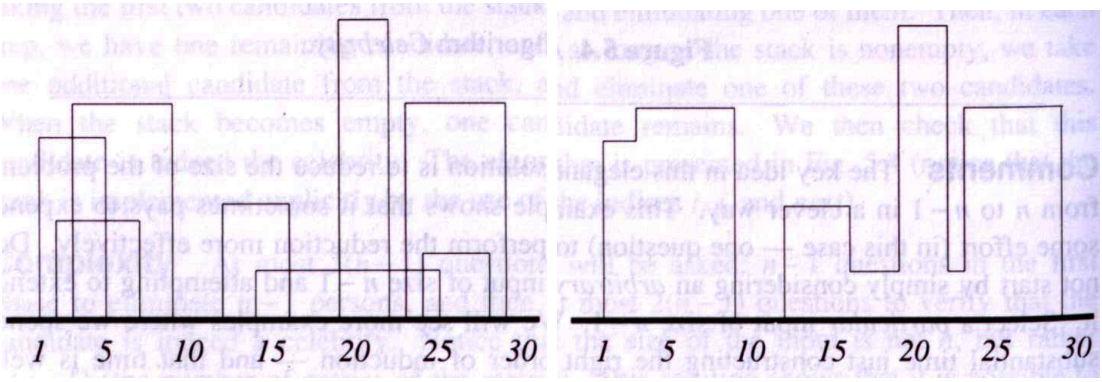


(b) The piece to be filled in the chessboard



(c) The chessboard covered by the pieces

Figure 1: The chessboard tiling problem



(a) The buildings (b) The skyline of the buildings

Figure 2: The buildings' skyline problem

Problem 4. Solve the following problems on the textbook:

1. (3%) 4.4-5 on p.93
2. (3%) 4.4-9 on p.93
3. (4%) 4.5-1 on p.96 (all 4 sub-questions)

Problem 5. (20%) In this problem, you are given the exact locations and shapes of several rectangular buildings in a city. Determine the skyline (in two dimensions) of the buildings and eliminate the hidden lines. For example, the buildings in Figure 2(a) correspond to the following input: (1,**11**,5), (2,**6**,7), (3,**13**,9), (12,**7**,16), (14,**3**,25), (19,**18**,22), (23,**13**,29), (24,**4**,28). The skyline in Figure 2(b) is represented as follows:
(1,**11**,3,**13**,9,**0**,12,**7**,16,**3**,19,**18**,22,**3**,23,**13**,29,**0**). Derive an algorithm using the divide-and-conquer strategy and analyze the running time of your algorithm.