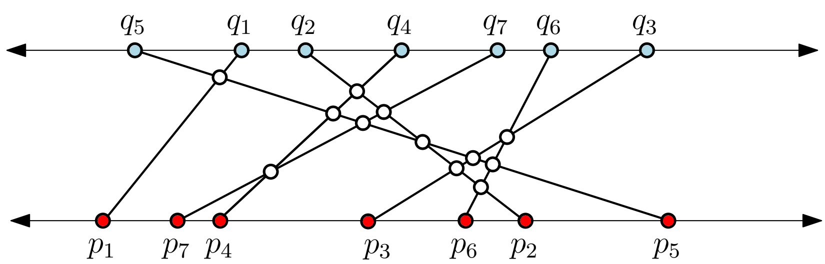
Homework 2 Due: 11:59pm, March 29, 2023

Problem 1 Suppose you are given a sorted array of n distinct numbers that has been *rotated* k steps, for some unknown integer k between 1 and $n-1$. That is, you are given an array $A[1,\ldots,n]$ such that some prefix $A[1,\ldots,k]$ is sorted in increasing order, the corresponding suffix $A[k+1,\ldots,n]$ is sorted in increasing order, and $A[n] < A[1]$. For example, you might be given an array $[8, 10, 12, 14, 19, 21, 39, 1, 3, 4, 5]$ (where $n = 11, k = 7$).

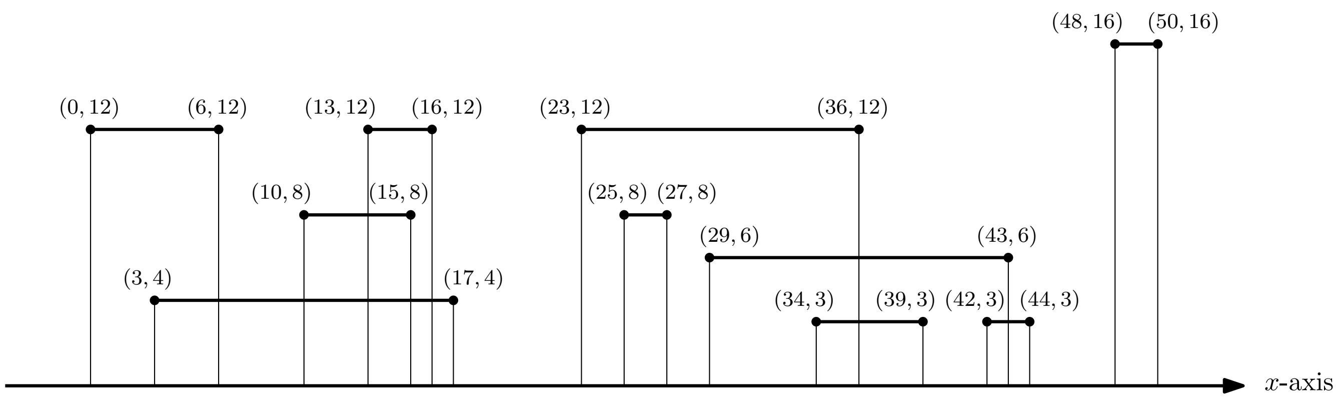
(a) (2 pts) Design a recursive algorithm to compute the unknown integer k in $O(\log n)$ time.

(b) (2 pts) Design a recursive algorithm to determine if the given array contains a given number x in $O(\log n)$ time.

Problem 2 (3 pts) Suppose you are given two sets of n points, one set $\{p_1, p_2, \ldots, p_n\}$ on the line $y = 0$ and the other set $\{q_1, q_2, \ldots, q_n\}$ on the line $y = 1$. Create a set of n line segments by connecting each point p_i to the corresponding q_i. Describe and analyze a recursive algorithm to determine how many pairs of these line segments intersect, in $O(n \log n)$ time.



Problem 3 (4 pts) Consider a set D of n axis-parallel rectangles lying above the x-axis whose bottom sides are all contained in the x-axis in the plane. Design an efficient recursive algorithm that computes the union of the rectangles in D, and analyze the running time of your algorithm. Each rectangle is given by the left and right endpoints (their x- and y-coordinates) of its top side (thick segment). The output, for each connected component C of the union, must contain a list of the vertices in order along the boundary of C.



Input: $((13, 12), (16, 12)), ((25, 8), (27, 8)), ((3, 4), (17, 4)), ((48, 16), (50, 16)), ((42, 3), (44, 3)), ((10, 8), (15, 8)), ((0, 12), (6, 12)), ((23, 12), (36, 12))$ $((34, 3), (39, 3)), ((29, 6), (43, 6))$.

Output:

1. $(0, 0), (0, 12), (6, 12), (6, 4), (10, 4), (10, 8), (13, 8), (13, 12), (16, 12), (16, 4), (17, 4), (17, 0)$.
2. $(23, 0), (23, 12), (36, 12), (36, 6), (43, 6), (43, 3), (44, 3), (44, 0)$.
3. $(48, 0), (48, 16), (50, 16), (50, 0)$.

Homework 2 Due: 11:59pm, March 29, 2023

Problem 4

(a) (2 pts) Let $A[1,\ldots,m]$ and $B[1\ldots,n]$ be two arbitrary arrays. A *common subsequence* of A and B is both a subsequence of A and a subsequence of B. Give a backtracking algorithm(a recurrence relation) that gives the length of the *longest* common subsequence of A and B.

(b) (2 pts) Let $A[1,\ldots,m]$ and $B[1\ldots,n]$ be two arbitrary arrays. A *common supersequence* of A and B is another sequence that contains both A and B as subsequences. Give a backtracking algorithm(a recurrence relation) that gives the length of the *shortest* common supersequence of A and B.

Problem 5 (3 pts) Let $X[1,\ldots,k]$ and $Y[1\ldots,n]$ be two arbitrary arrays where $k \leq n$. Decribe a backtracking algorithm(a recurrence relation) to find the smallest number of symbols that can be removed from Y so that X is no longer a subsequence.

Problem 6 (3 pts) Describe a backtracking algorithm (a recurrence relation) that for a positive integer n, computes a shortest sequence of numbers $x_0 < x_1 < x_2 < \cdots < x_\ell = n$ satisfying the following rule.

- $x_0 = 1$.
- For every index $k > 0$, there are indices i, j with $i \leq j < k$ such that $x_k = x_i + x_j$.