FINAL EXAM

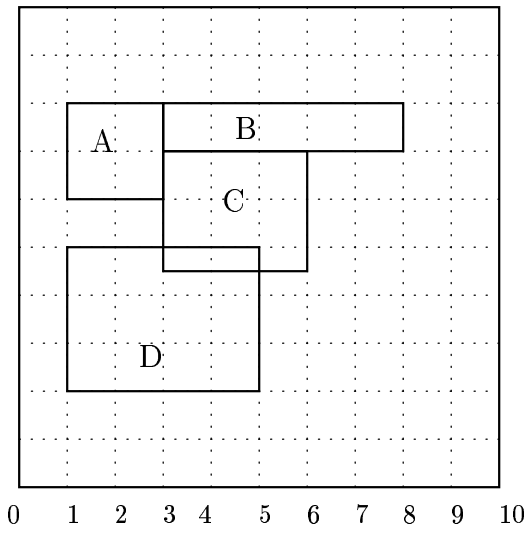
CMSC 420 – Spring 1992

Question 1. (15 points) Consider the set of rectangles shown in the figure below. You are required to:

(a) (3 point) List the x-coordinates at which a vertical sweep line will halt.

(b) (5 points) Specify the list of *active* rectangles when the vertical sweep line halts for the second time.

(c) (7 points) Specify, in 10 lines or less, what happens after the vertical sweep line halts in (b) above. Be careful to specify the precise *sequence* in which events occur. You will be penalized if your answer exceeds 10 lines.



Question 2. (25 points) Consider the following problem: you have a pointer T to the root of a segment tree. The segment tree represents line segments lying along a horizontal line. You are given a point p (i.e. a real number on this horizontal line). You are supposed to find the set of *all* lines represented in the segment tree that contain this point.

(a) (5 points) Specify, in PASCAL-like notation, a data structure for the nodes in the segment tree.

(b) (20 points) Develop an algorithm to solve the above problem. The algorithm takes as input, a pointer T, to the root of a segment tree whose nodes have the structure specified in (a), and a point p.

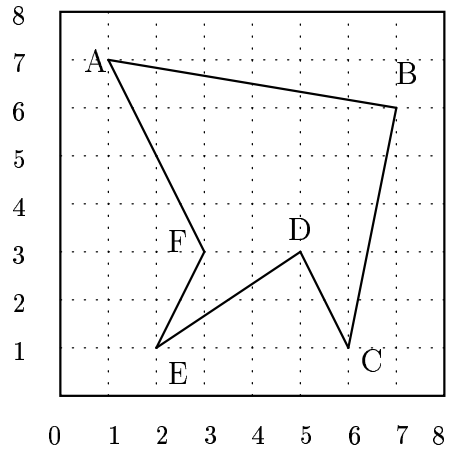
Question 3. (25 points) How would the set of lines in the following figure (assume they are inserted in the order: AB,BC,CD,DE,EF,FA) be represented using:

(a) (7 points) A PM_1-quadtree ?

(b) (7 points) A PM_2-quadtree ?

(c) (6 points) Explain, in 5 lines or less, the circumstances under which, given a set S of lines to be represented, the PM_1-quadtree representing S has a smaller height than the PM_2-quadtree representing S. You will be penalized if you provide an answer that is over five lines long.

(d) (5 points) How does the order in which individual lines are inserted into a PM-quadtree affect the final structure of the PM-quadtree ?



Question 4. (10 points) Suppose we have 512 memory locations (addresses 0 through 511) available for dynamic storage allocation, and that the buddy system will be used to dynamically allocate/liberate this memory. Suppose that the transactions listed below occur one after the other in the order in which they are listed.

1. Transaction 1: a process $P1$ requests 128 locations of memory.
2. Transaction 2: another process $P2$ requests 64 locations of memory.
3. Transaction 3: process $P3$ requests 128 locations of memory.
4. Transaction 4: process $P1$ returns its 128 locations of memory.
5. Transaction 5: Process $P3$ returns its 128 locations of memory.

You are required to specify the following information at the end of each transaction (i.e. once each of transactions 1 through 5 above are completed).

1. For each chunk of memory that is NOT free (i.e. has been allocated to some process), state: (a) the starting location of the chunk of memory, (b) the size of the chunk of memory, (c) the process to which it has been allocated and (d) the starting address of its buddy.
2. For each free chunk of memory, state: (a) the starting location of the chunk of memory, and (b) the size of the chunk and (c) the starting address of its buddy.

Question 5. (25 points) Suppose you are given the sequence of integers

$$41, 5, 17, 58, 72, 46, 20.$$

You are required to:

(a) (2 points) Construct a binary search tree to represent these integers. Assume the values are inserted into the tree in the order in which they have been listed.

(b) (6 points) Traverse the tree in post-order. List the post-order traversal of the tree.

(c) (5 points) Thread the tree in post-order.

(d) (12 points) Suppose you are given a pointer to a node N in the tree. Find the *predecessor* of N in post-order. If there is no such predecessor, return NIL.