# Plane-Sweep and Rectangle Intersection problems

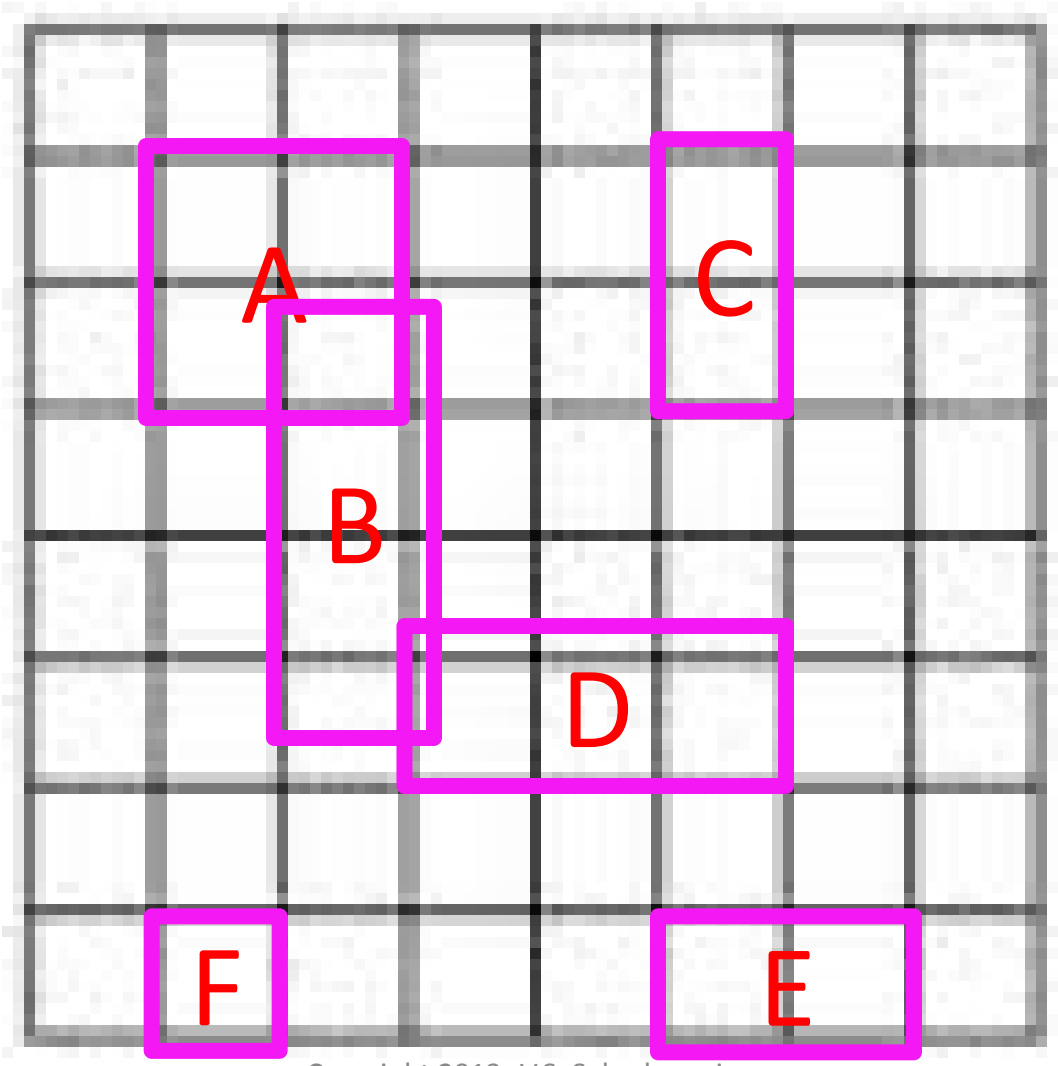
V.S. Subrahmanian

Spring 2013

# The Problem

- INPUT:
  - A set **R** of rectangles. Each rectangle is represented by 4 numbers corresponding to the *x*-coordinate of its left and right sides, and the *y*-coordinate of its bottom and top sides. You can assume all the rectangles are "closed" on all sides.
- OUTPUT:
  - Return all pairs of rectangles that intersect.

# Example

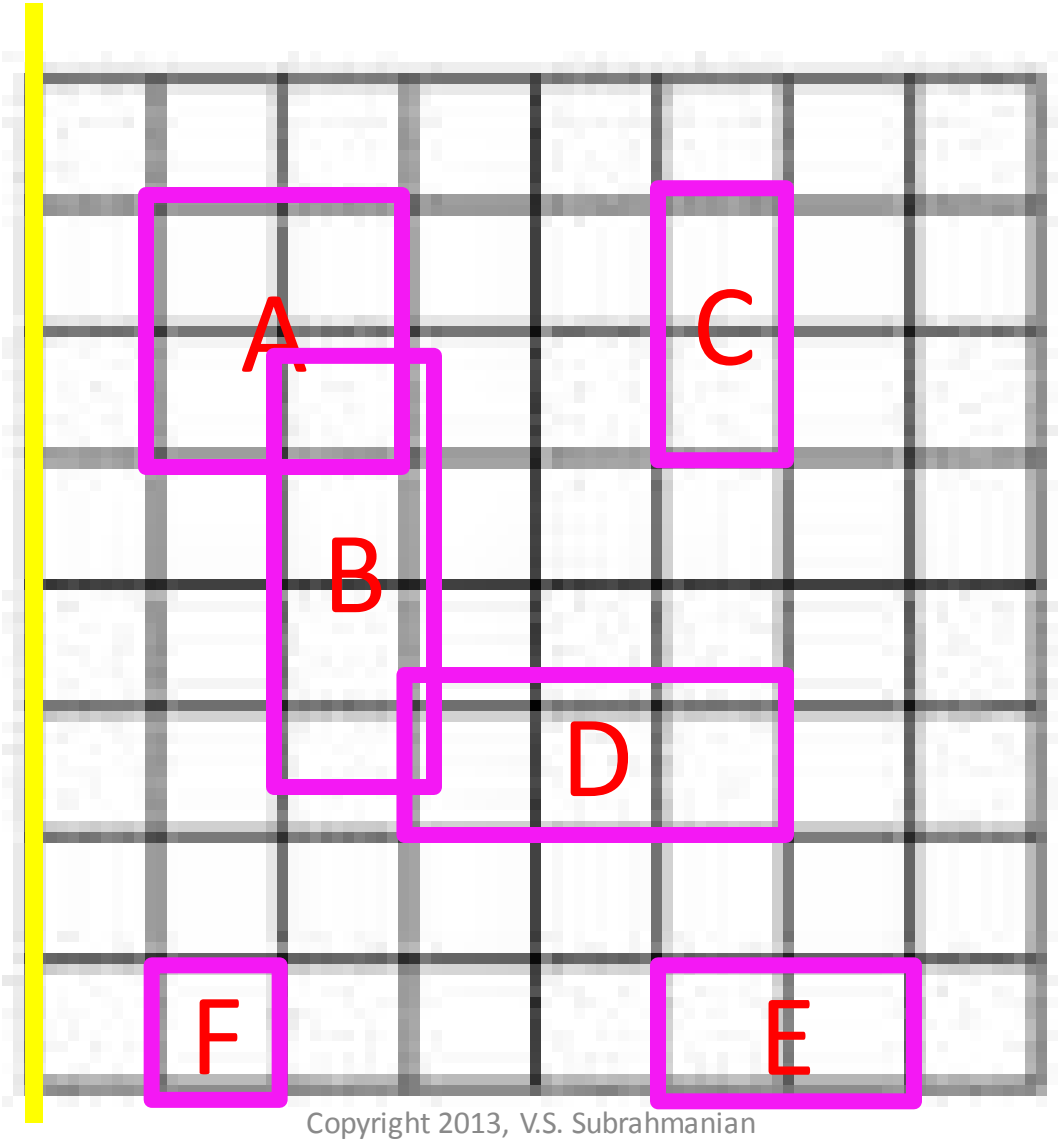


# Basic Idea

- Move a vertical “sweep” line from left to right across the 2-d space.
- The line “stops” whenever the sweep line either
  - “enters” a rectangle or
  - “exits” a rectangle
- The algorithm maintains an “active list” of rectangles which is updated when the sweep line steps .
- The active list is the set of rectangles that have been “entered” but not “exited” by the sweep line.
- We check for rectangle intersection only when a rectangle is added to the active list by comparing it with rectangles already in the active list.

# Example

Sweep line starts at 0. ACTIVE and ANSWER are both initialized to NIL.



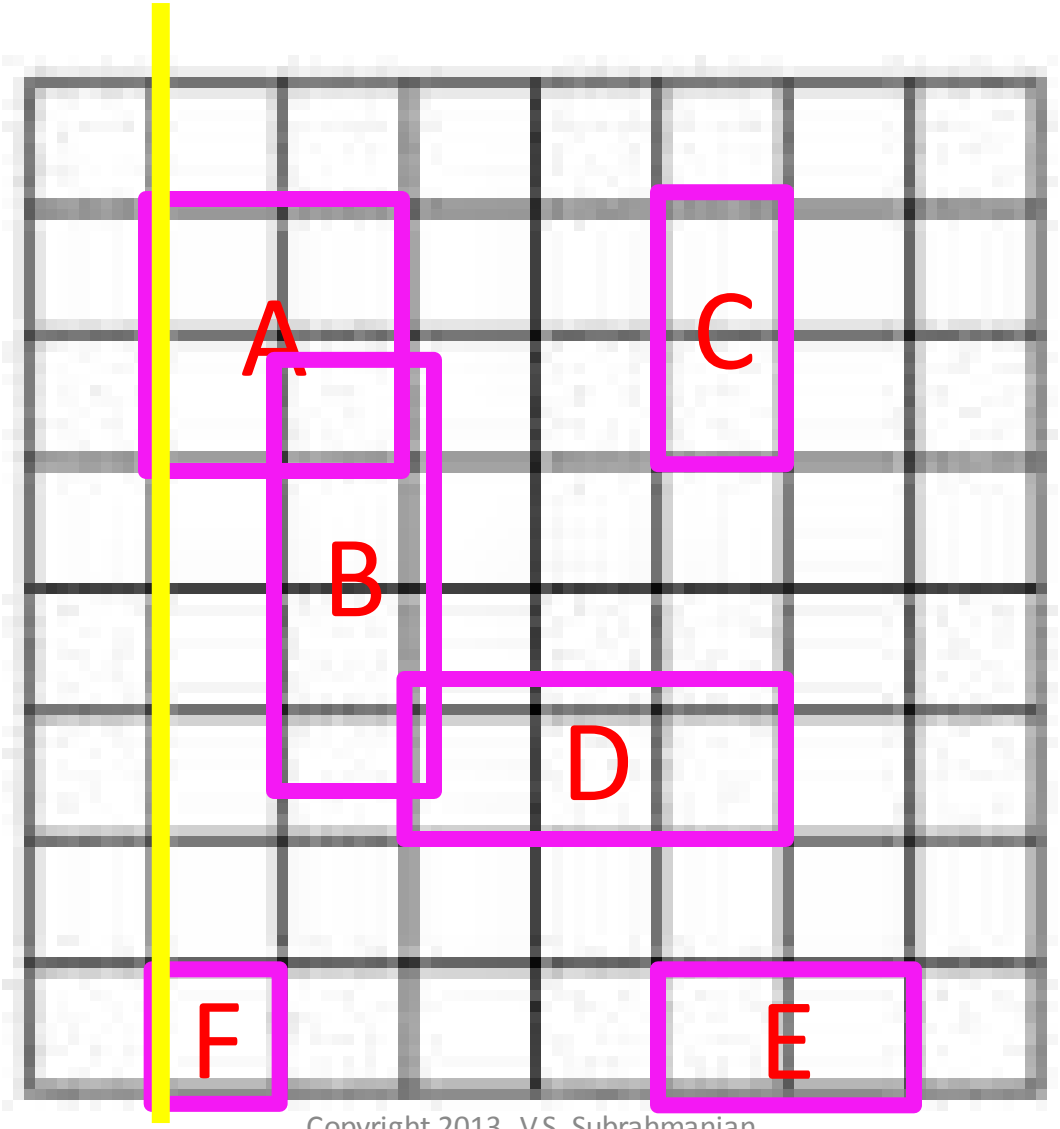

ACTIVE

ANSWER

# Example

Sweep line moves to the location shown. Rectangles A,F are entered.

- Insert A into ACTIVE. Check if A intersects any rectangle in ACTIVE. No.
- Insert F into ACTIVE. Check if F intersects any rectangle in ACTIVE. No.



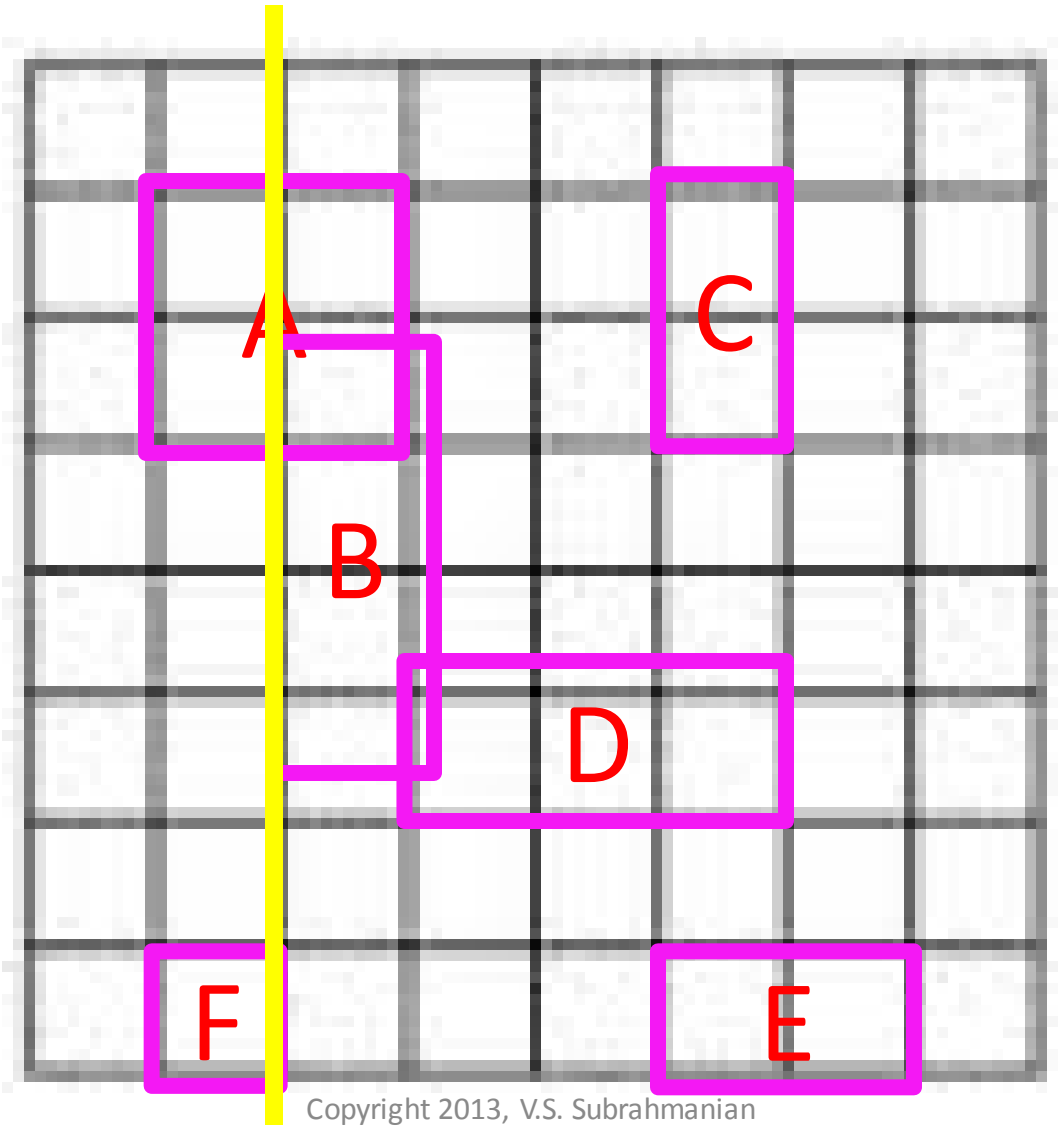

ACTIVE
{A,F}

ANSWER

# Example

Sweep line moves to the location shown. Rectangle B is entered, F is exited.

- Insert B into ACTIVE. Check if B intersects any rectangle in ACTIVE.
- It intersects A.
- So insert (A,B) into ANS.
- Delete F.



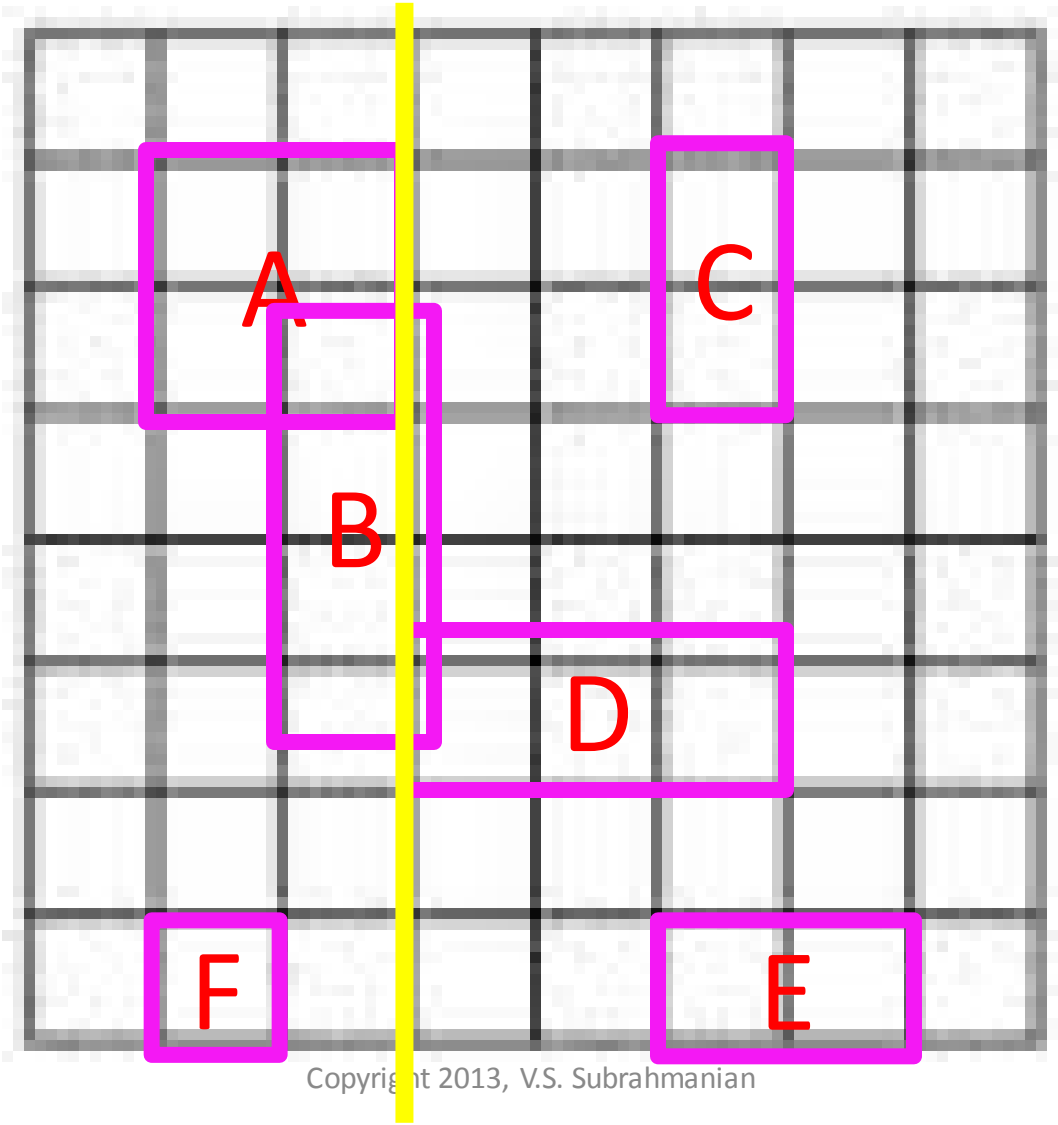

**ACTIVE**
**{A,B}**

**ANSWER**
**{ (A,B) }**

# Example

Sweep line moves to the location shown. Rectangle D is entered, A is exited.

- Insert D into ACTIVE. D intersects B, so put (B,D) into ANSWER.
- Delete A from active list.



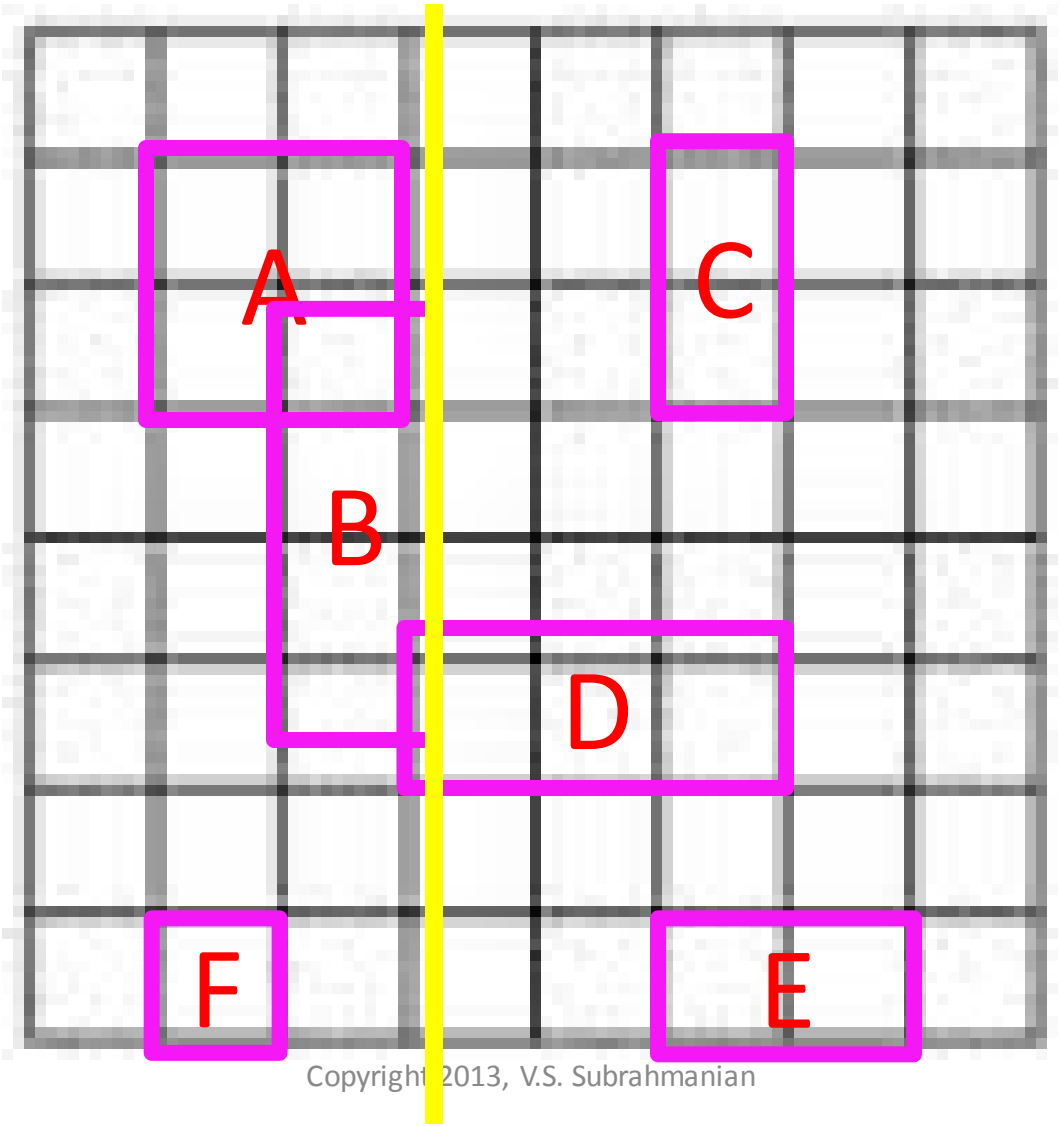

**ACTIVE**
**{B,D}**

**ANSWER**
**{ (A,B), (D,B)}**

# Example

Sweep line moves to the location shown. Rectangle B is exited.

- Delete B from active list.



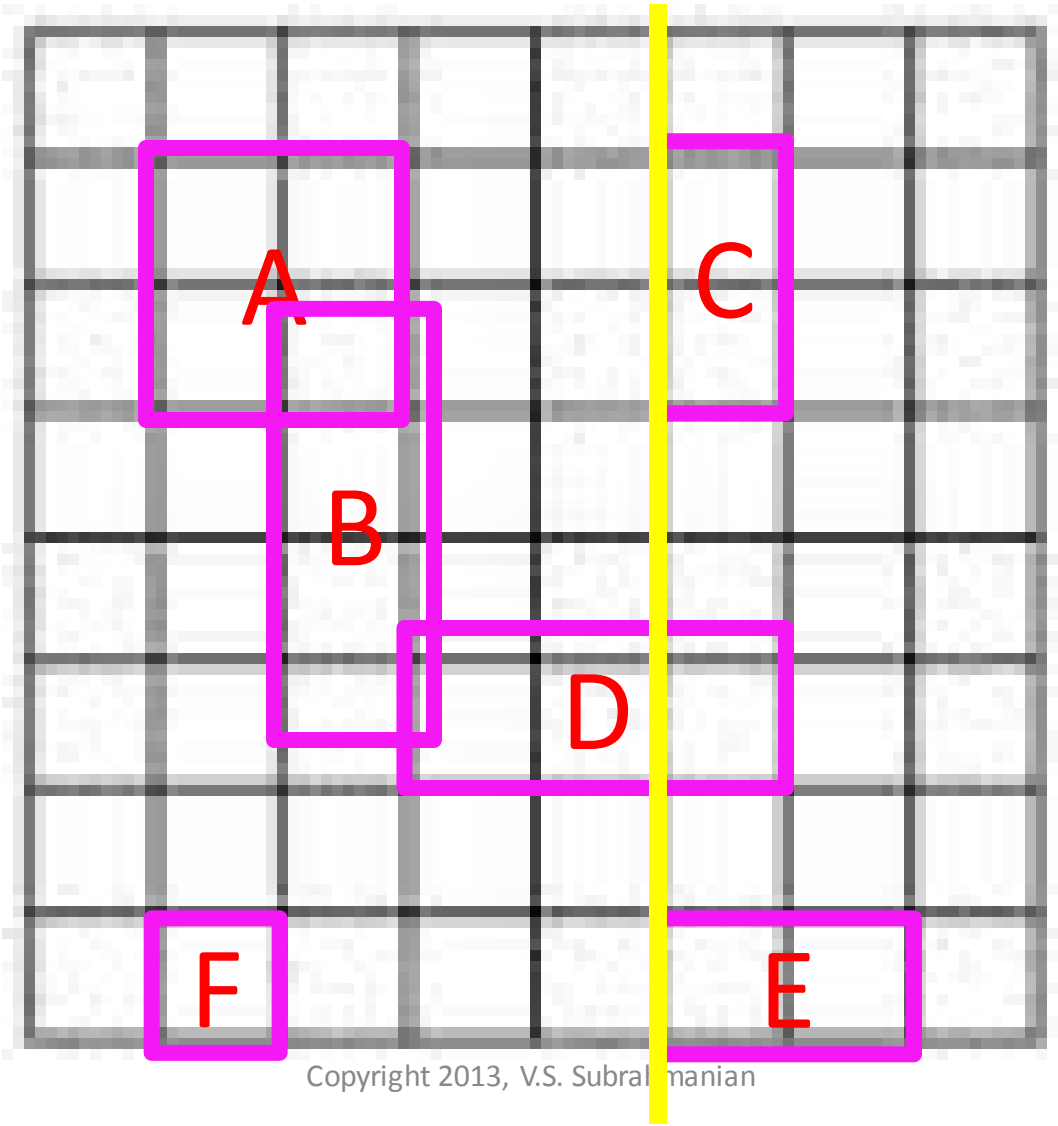

**ACTIVE {D}**

**ANSWER { (A,B), (D,B)}**

# Example

Sweep line moves to the location shown. Rectangles C,E are entered.

- Both C and E must be inserted into the active list.
- They do not intersect anything, so ANSWER stays unchanged.



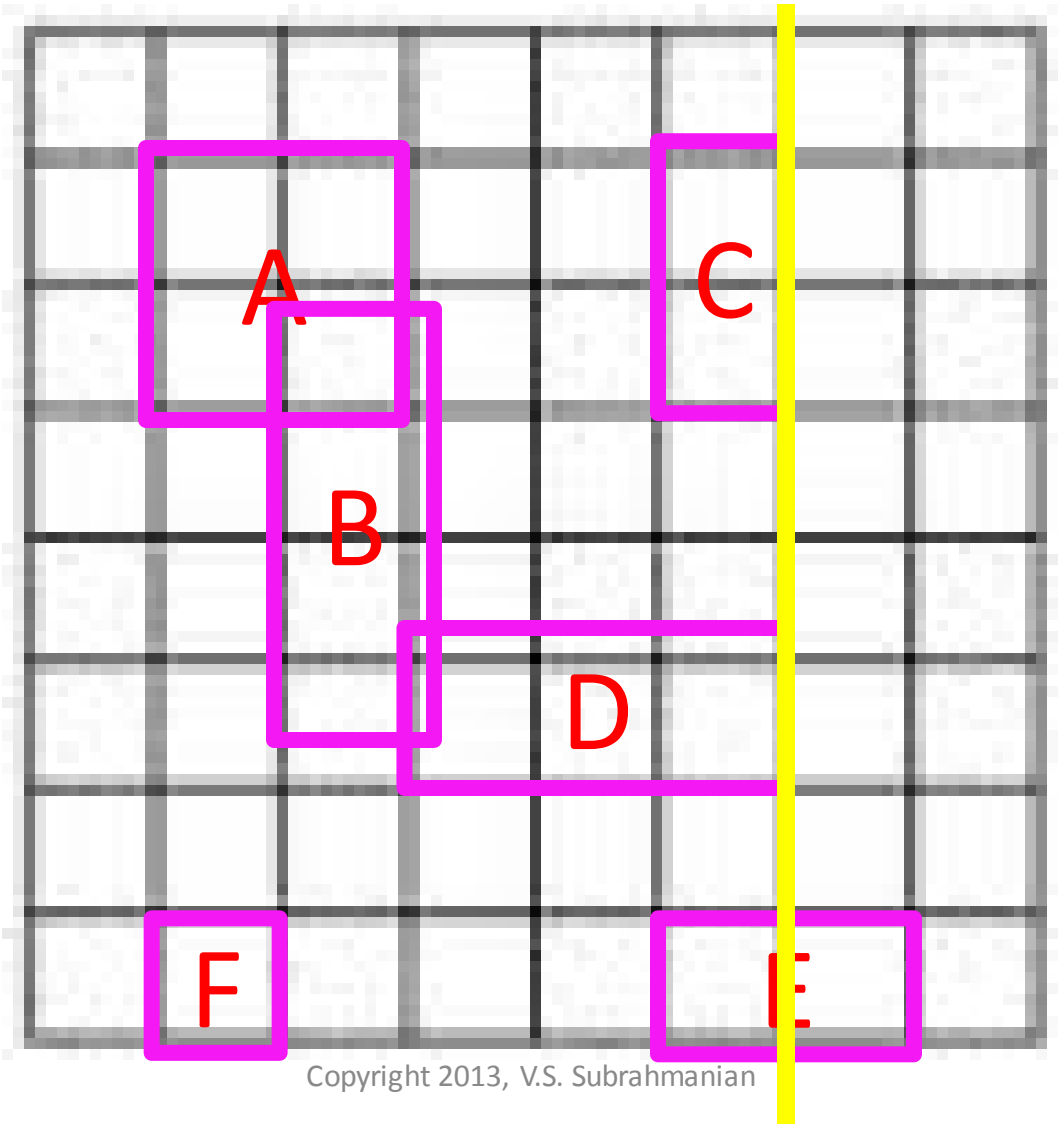

**ACTIVE**
**{C,D,E}**

**ANSWER**
**{ (A,B), (D,B)}**

# Example

Sweep line moves to the location shown. Rectangles C,D are exited.

- Both C and D must be deleted from the active list.



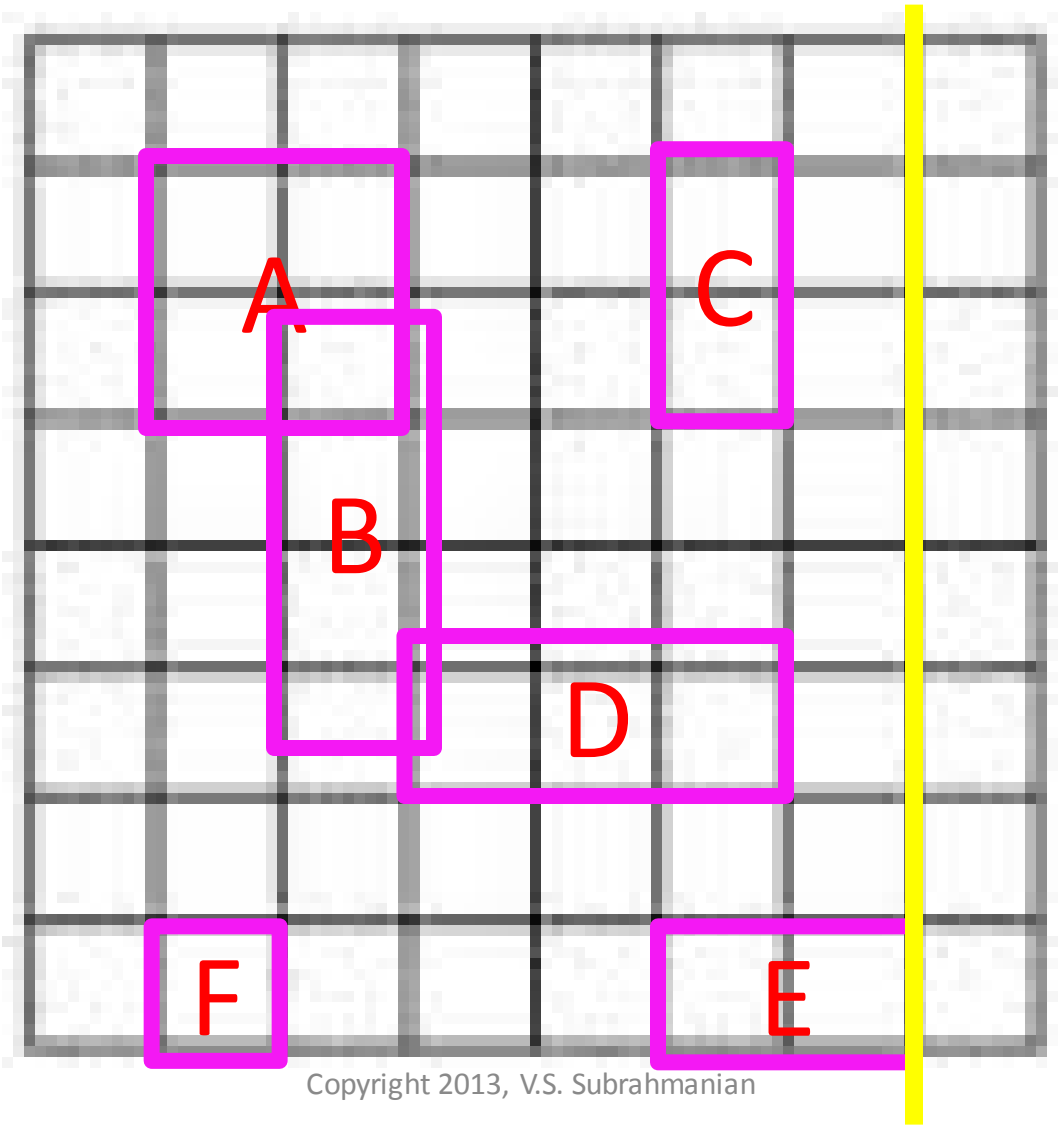

**ACTIVE {E}**

**ANSWER { (A,B), (D,B)}**

# Example

Sweep line moves to the location shown. Rectangle E is exited.

- E must be deleted from the active list.
- ACTIVE is NIL.
- No other stopping points for the algorithm.
- Return final answer.




**ACTIVE {E}**

**ANSWER { (A,B), (D,B)}**

# Details

- Rectangle data is represented by a table.

| RECTANGLE | LEFT | RIGHT | BOTTOM | TOP |
|---|---|---|---|---|
| A | | | | |
| B | | | | |
| C | | | | |
| D | | | | |
| E | | | | |
| F | | | | |
| G | | | | |

# Details

- Rectangle data is represented by a table.

| RECTANGLE | LEFT | RIGHT | BOTTOM | TOP |
|---|---|---|---|---|
| A | | | | |
| B | | | | |
| C | | | | |
| D | | | | |
| E | | | | |
| F | | | | |
| G | | | | |

**To find locations where the sweep line should STOP, merely sort the LEFT and RIGHT columns of the rectangle table in ascending order.**

# When Inserting into Active List

- Suppose we are checking to see if the the rectangle
  - rec = (L,R,B,T) intersects a rectangle
  - Rec' = (L',R',B',T') in the active list.
- As both rectangle as already "active", we just need to check if the intervals [B,T] and [B',T'] intersect, i.e. we need to check if

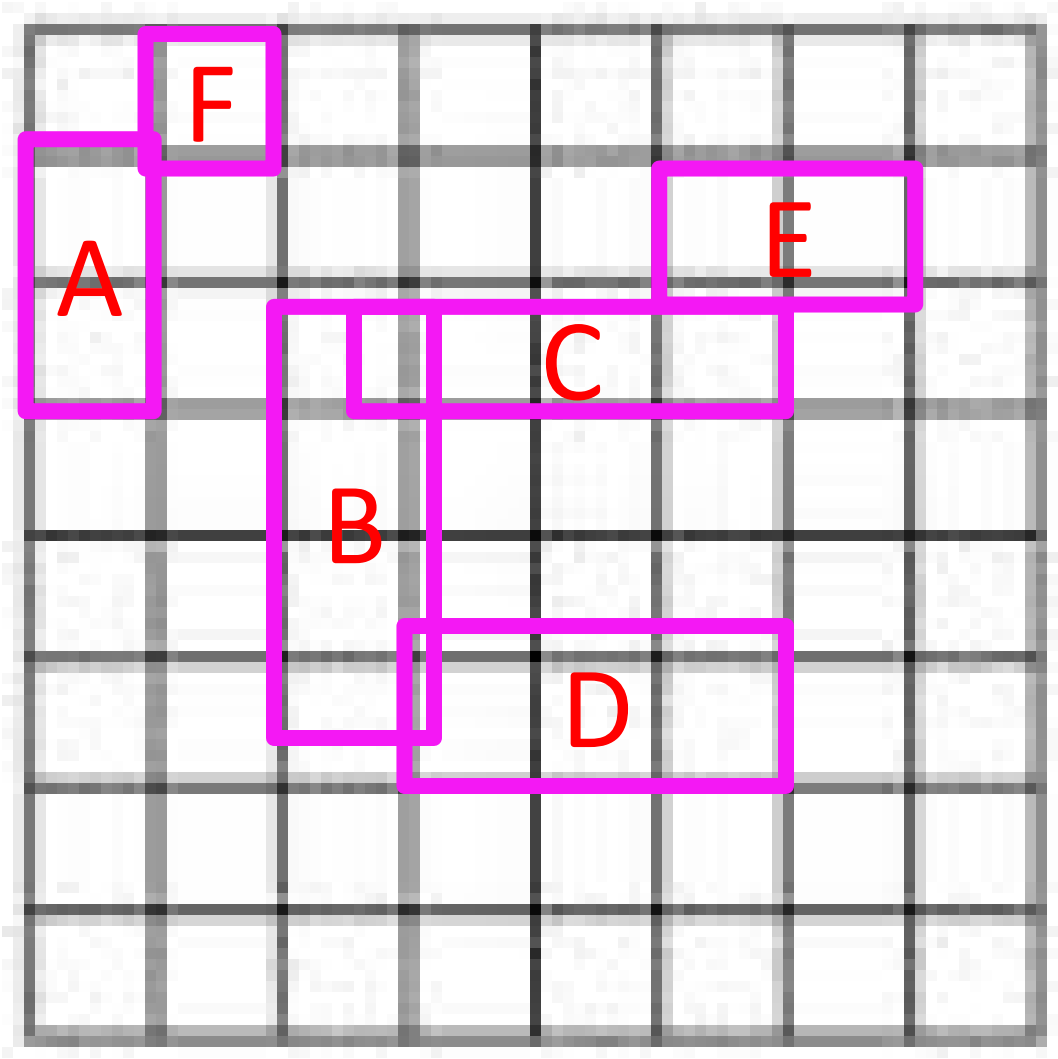
$$B \leq y \leq T$$
$$B' \leq y \leq T'$$

  is solvable. This is the case if and only if:

$$\max(B, B') \leq \min(T, T') .$$

- Thus, when checking whether a rectangle being inserted into the active list intersects a rectangle already there, we only need to check this.

# In class exercise 1



# In class exercise 2