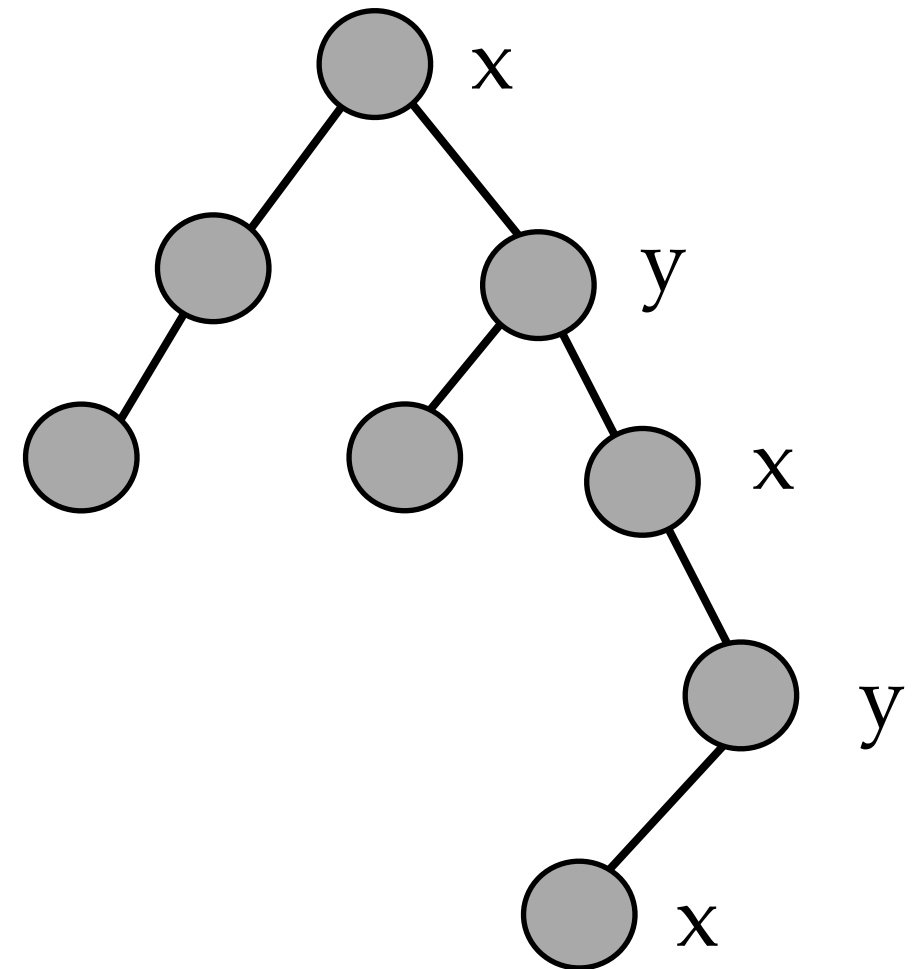


kd-Trees

- Invented in 1970s by Jon Bentley
- Name originally meant “3d-trees, 4d-trees, etc” where k was the # of dimensions
- Now, people say “kd-tree of dimension d ”
- Idea: Each level of the tree compares against 1 dimension.
- Let's us have only **two children** at each node (instead of 2^d)

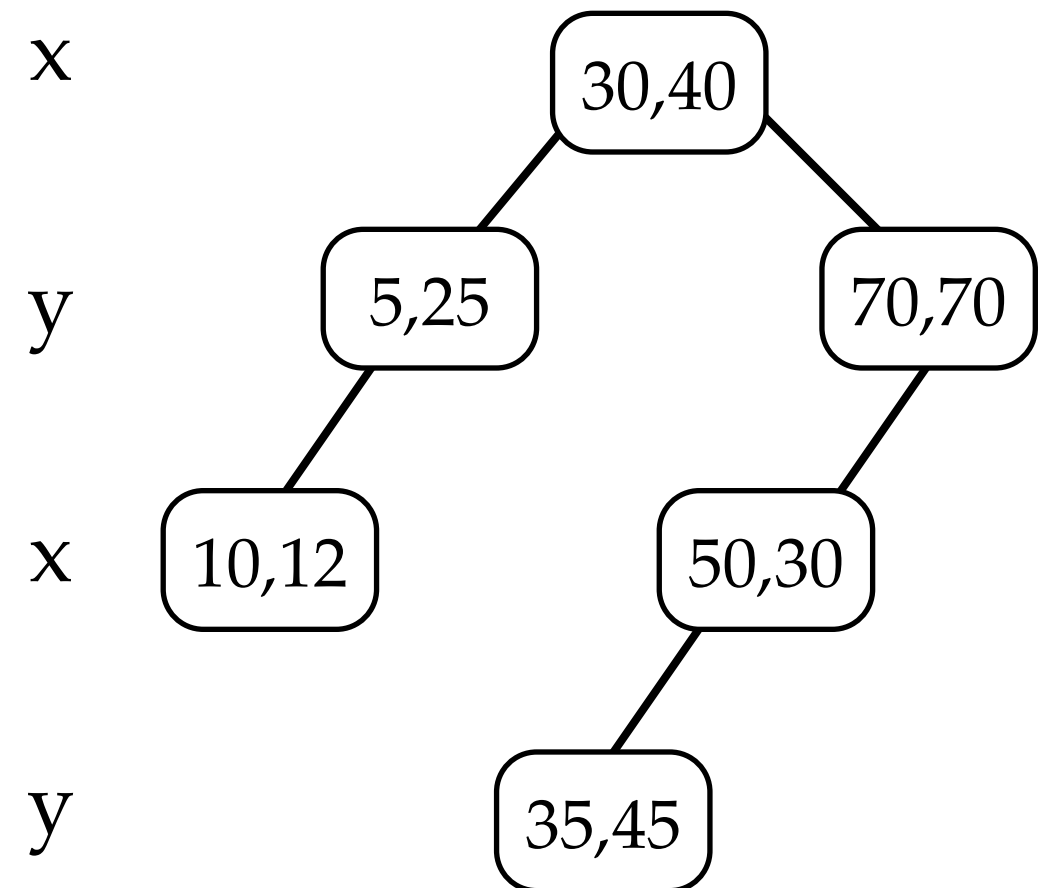
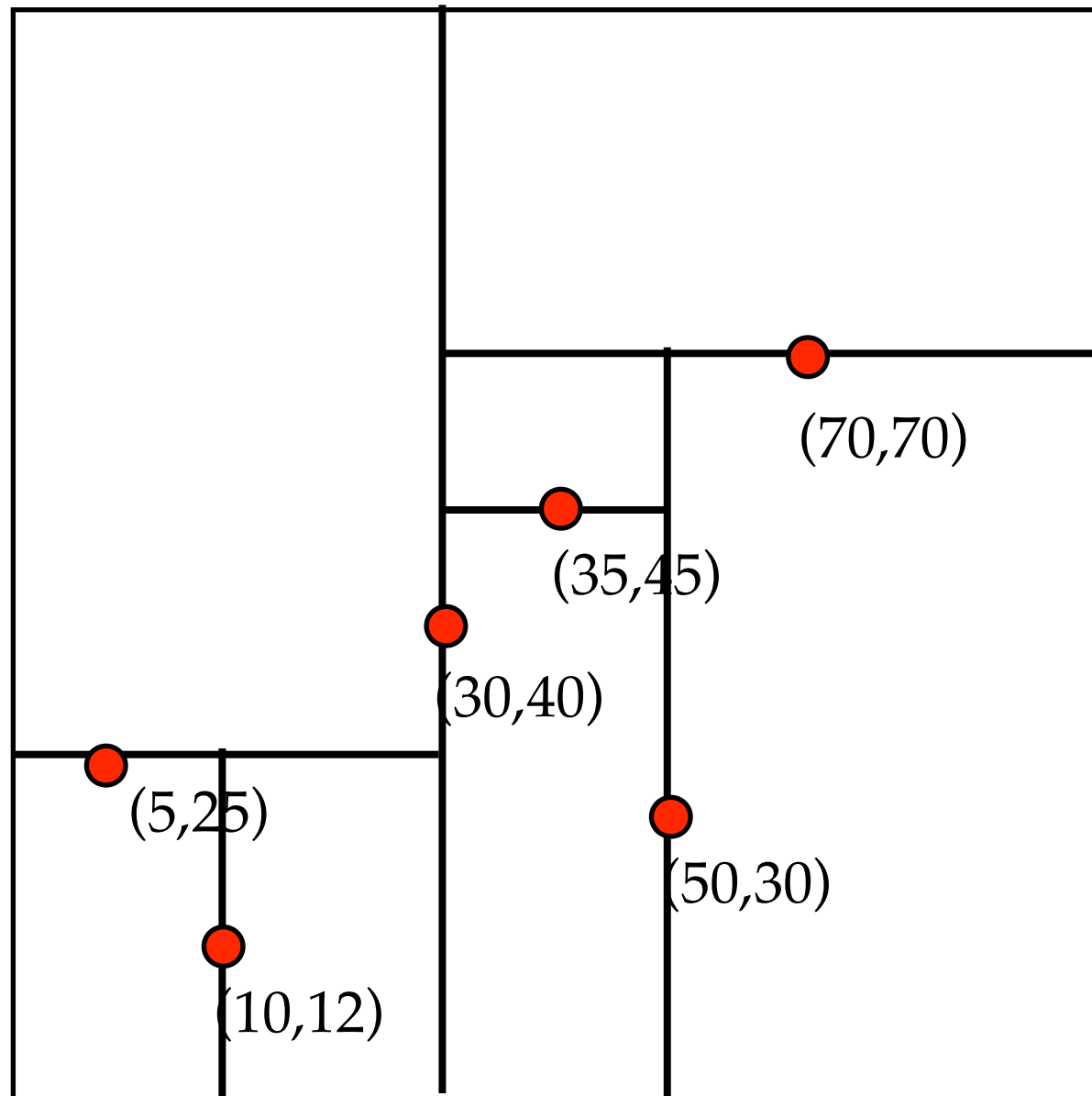
kd-trees

- Each level has a “cutting dimension”
- Cycle through the dimensions as you walk down the tree.
- Each node contains a point $P = (x, y)$
- To find (x', y') you only compare coordinate from the cutting dimension
 - e.g. if cutting dimension is x , then you ask: is $x' < x$?



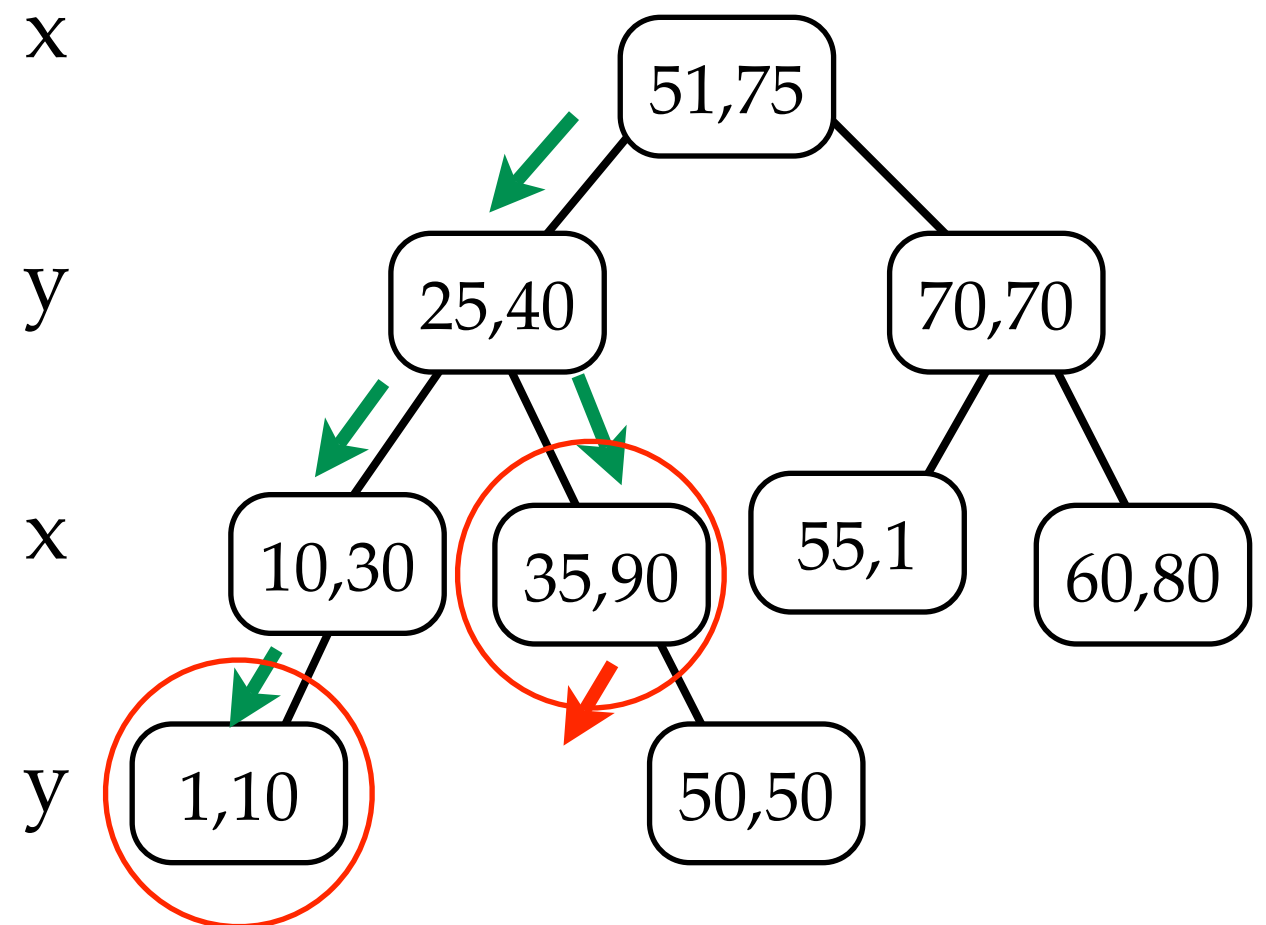
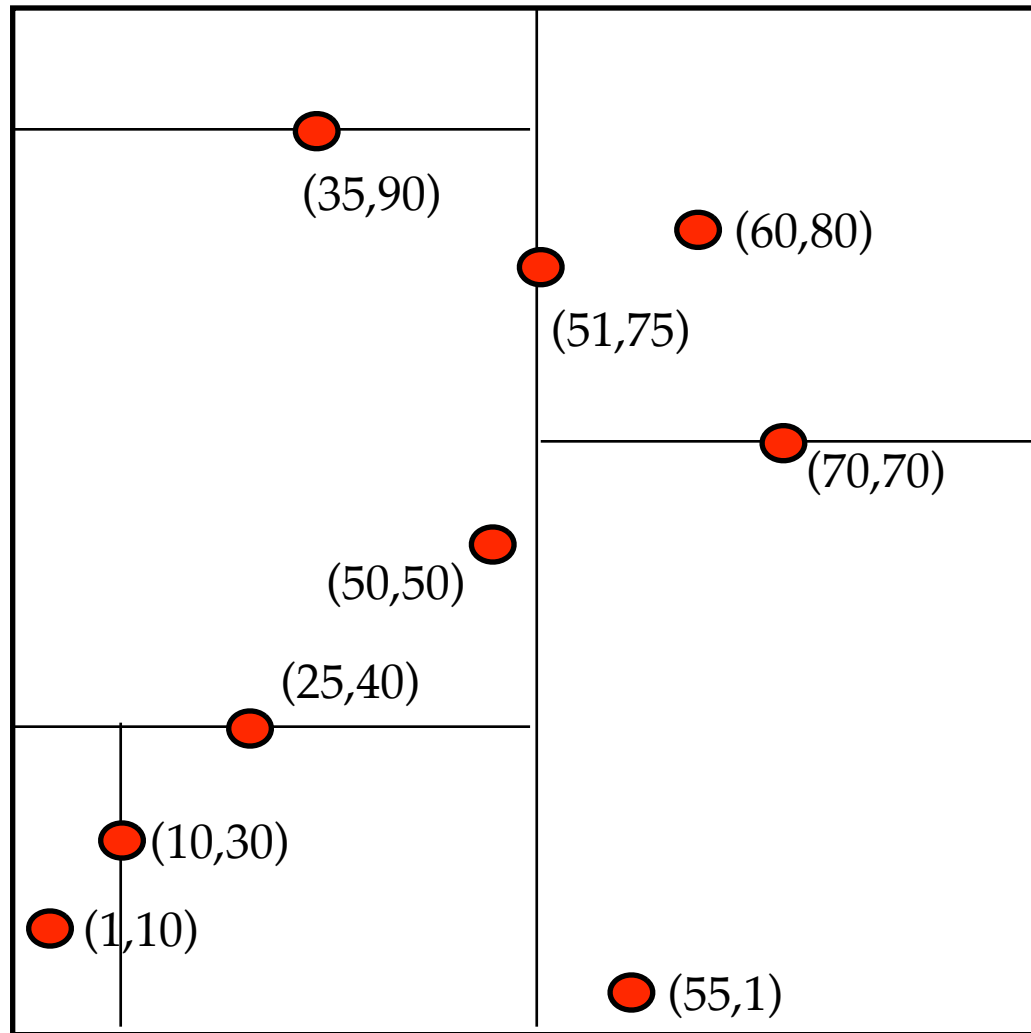
kd-tree example

insert: (30,40), (5,25), (10,12), (70,70), (50,30), (35,45)



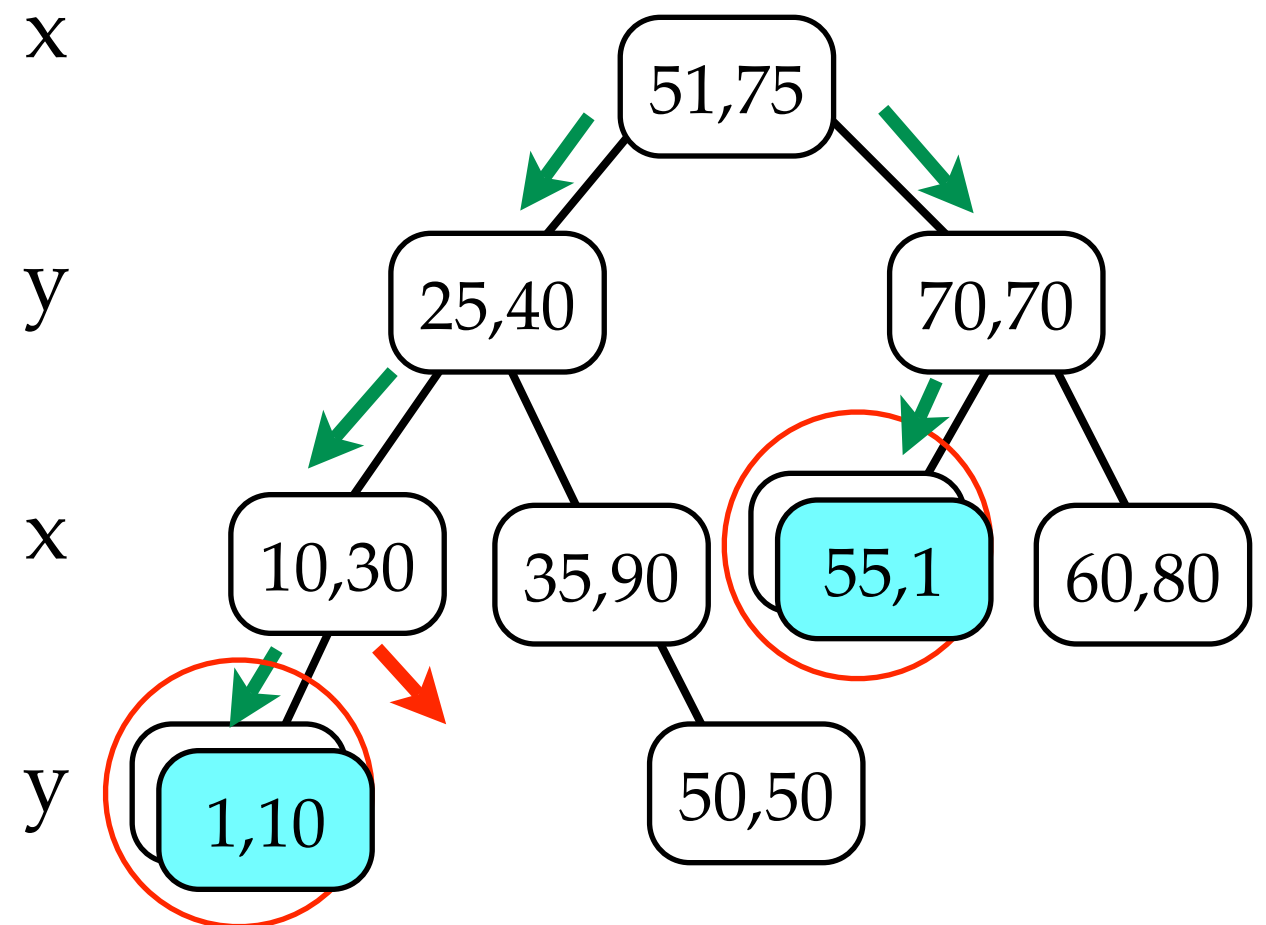
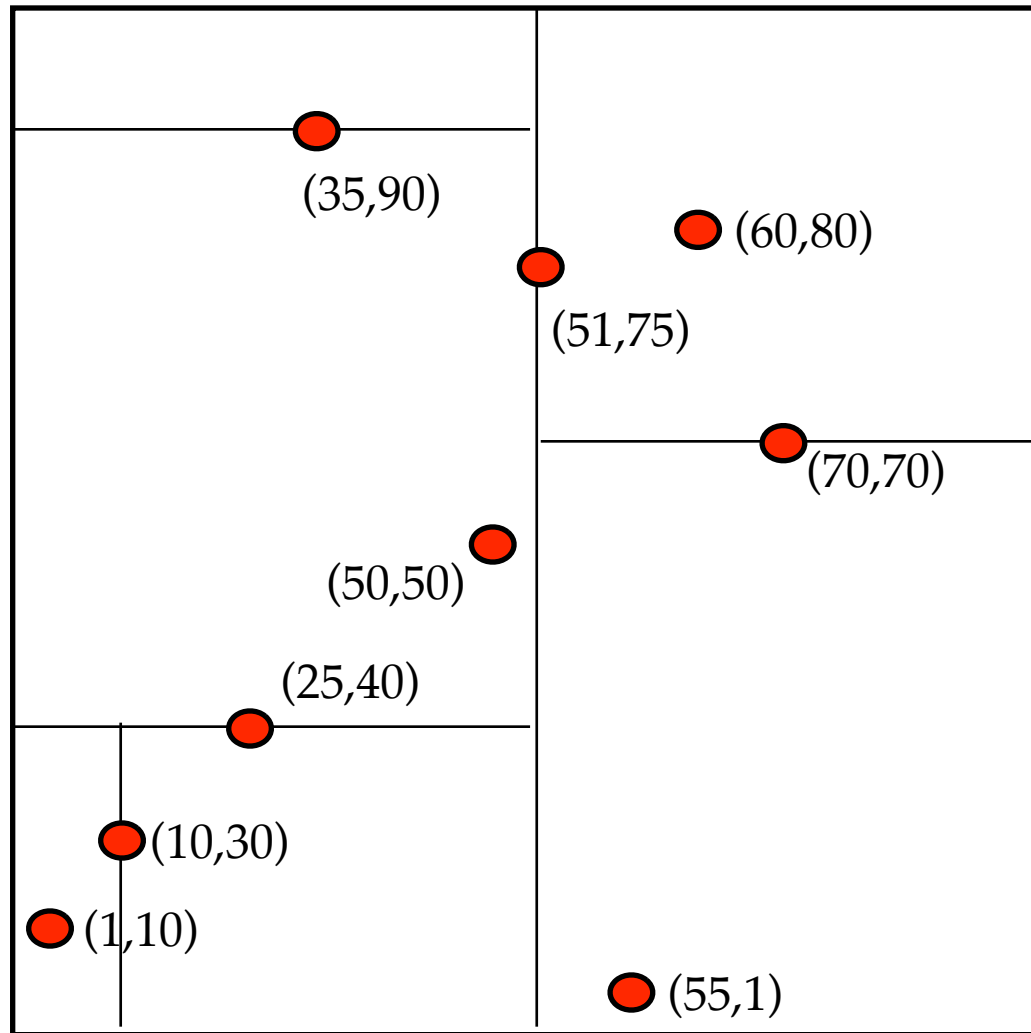
FindMin

FindMin(x-dimension):



FindMin

FindMin(y-dimension):



FindMin

FindMin(y-dimension): space searched

