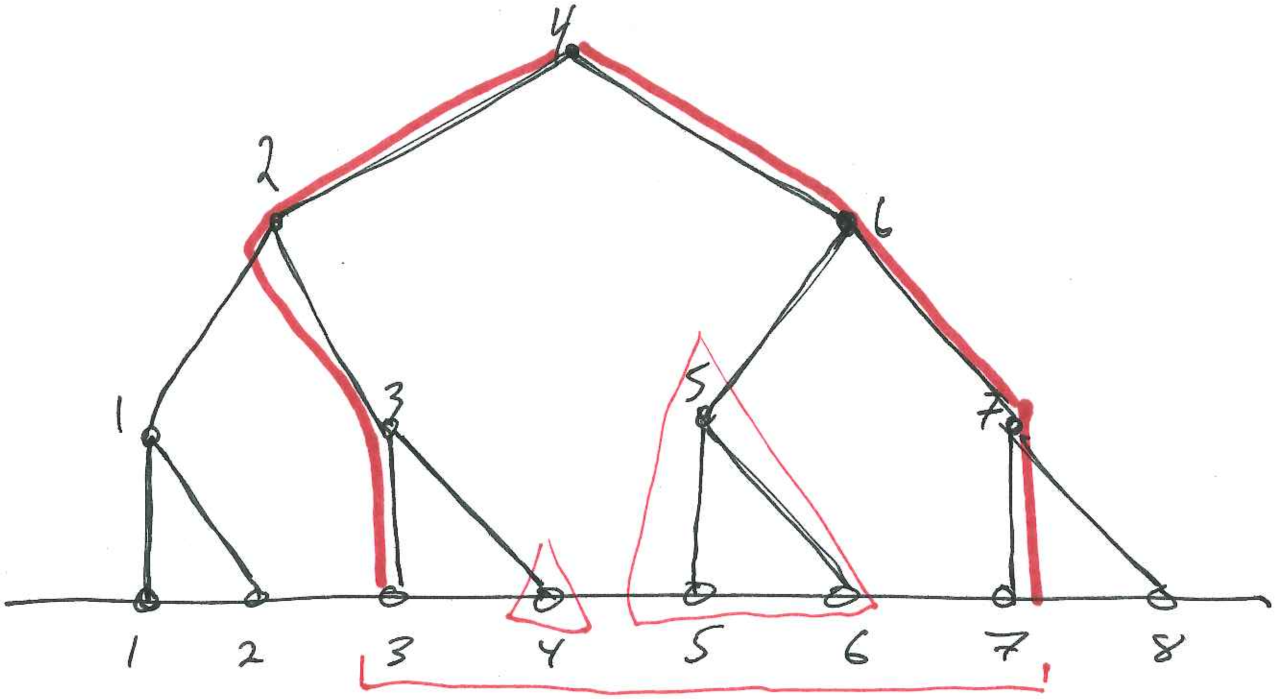
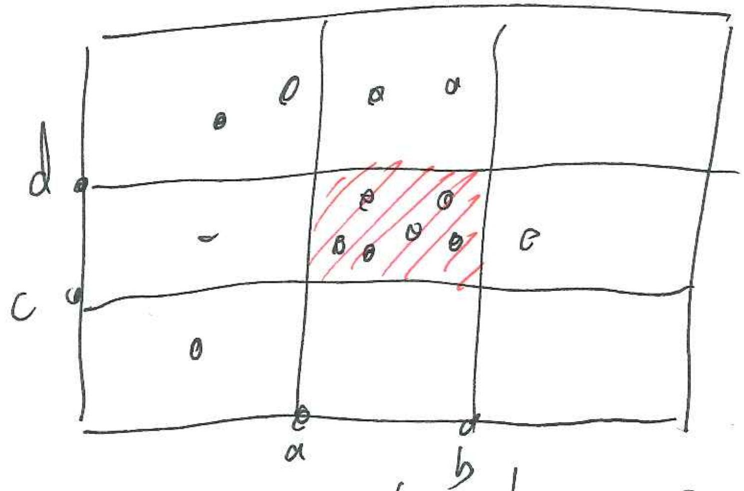
1 D Range search



2D Orthogonal Range Search



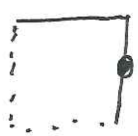
Queries: rectangles $[a,b] \times [c,d]$

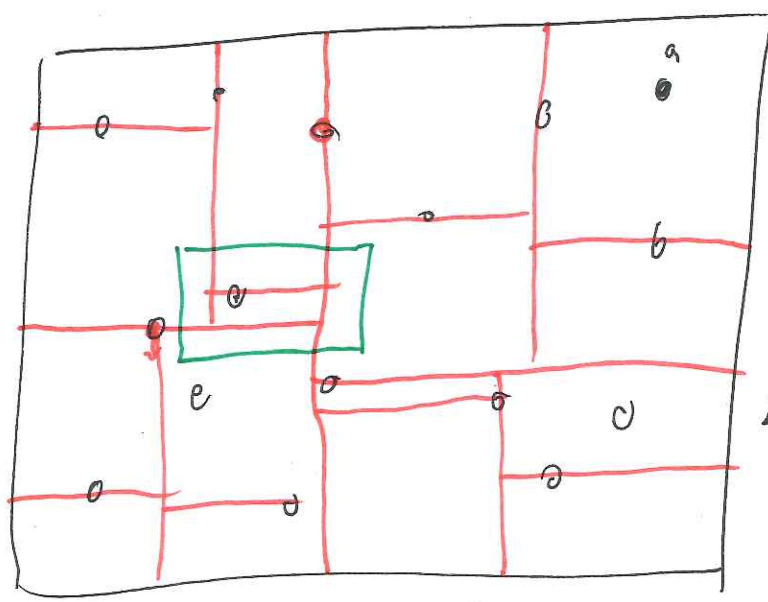
K-d-trees

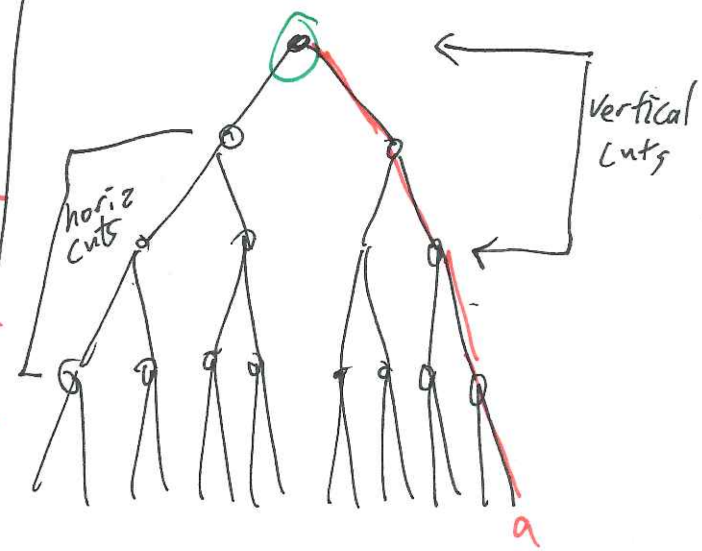
Idea: Recursively divide space,
alternating horiz. and vertical cuts

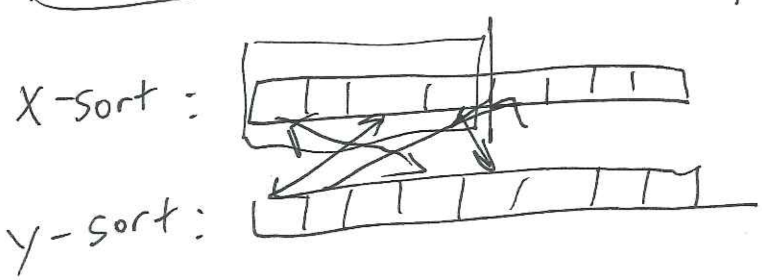
median ⌊n/2⌋ point

regions are open on left + bottom





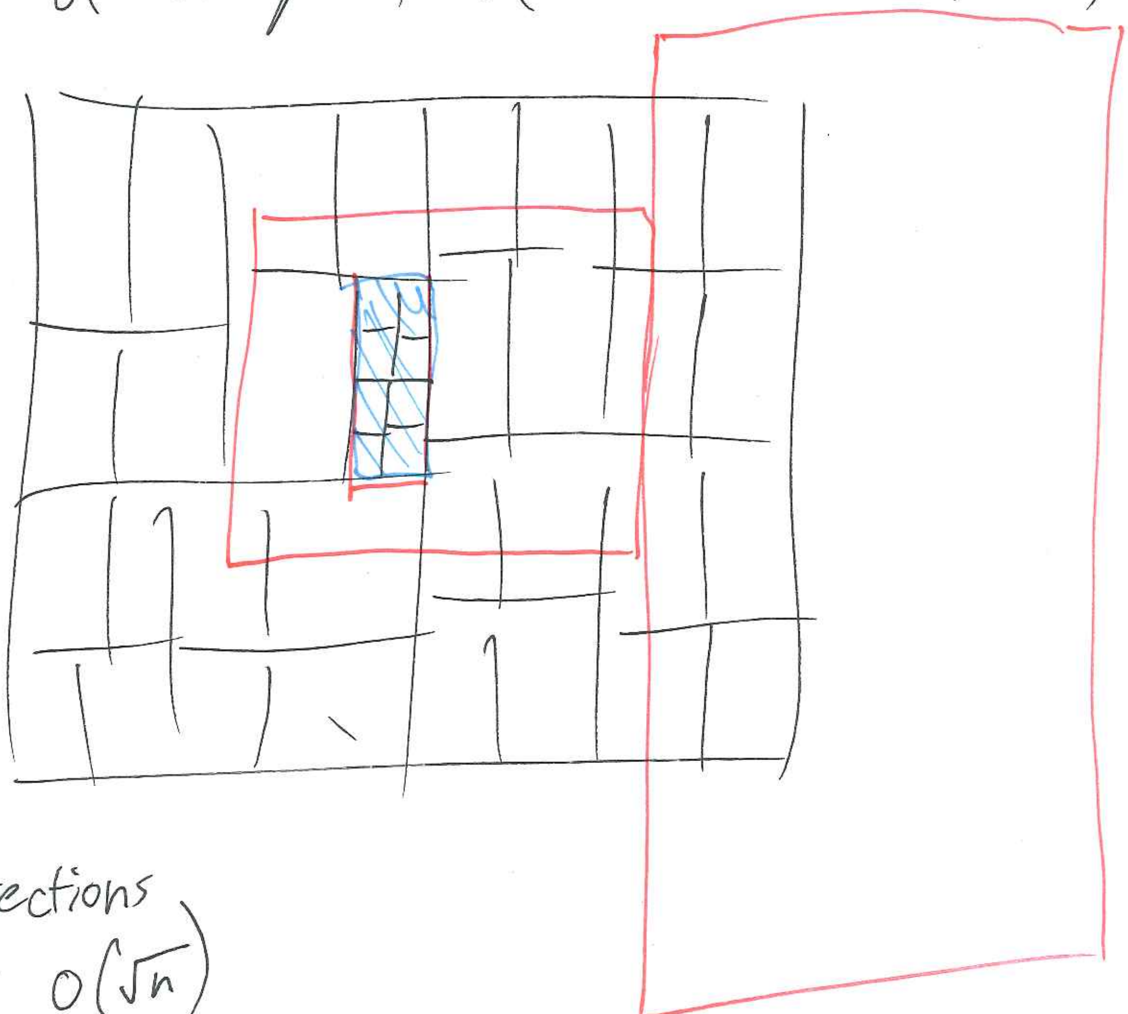




Query

```
Search(T, R)
  if (T is a leaf v) {return pt(v) if pt(v) ∈ R}
  else
    [ if region(T.left) ⊆ R
          then ReportSubtree(T.left)
      else if (region T.left ∩ R)
          then Search(T.left, R)
    ]
    [ Same for T.right
    ]
```

Cost of a query $O(k + \#\text{cell} \cap \text{bdy}(R))$



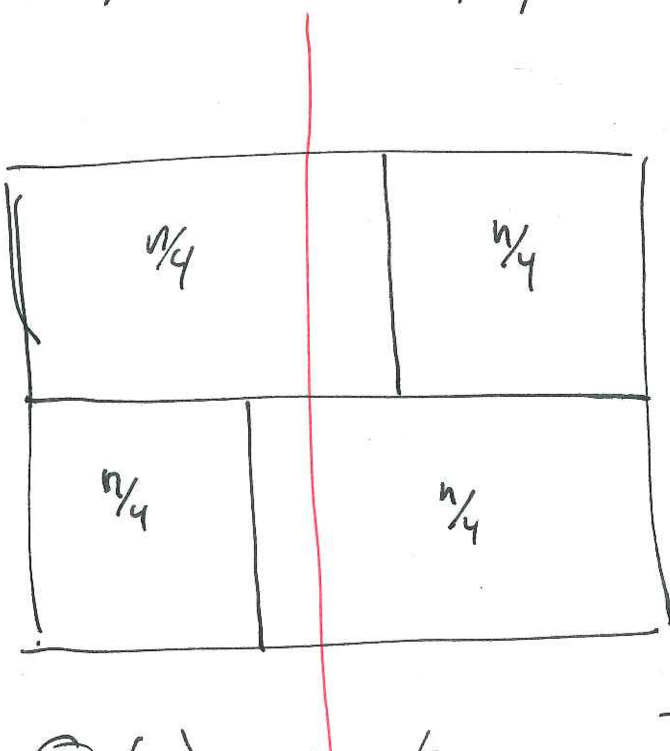
$2^{\frac{1}{2}(\log n)} = \sqrt{n}$

number of intersections of a vertical line w/ kd-tree.

$\Rightarrow$ total intersections w/ bdy is $O(\sqrt{n})$

cost $= O(k+\sqrt{n})$

Every two levels, the line can intersect two cells, each with $n/4$ points.



$$\left.\begin{aligned} Q(1) &= \text{const} \\ Q(n) &= 2 + 2Q\left(\tfrac{n}{4}\right) \end{aligned}\right\} \Rightarrow Q(n) = \sqrt{n}$$