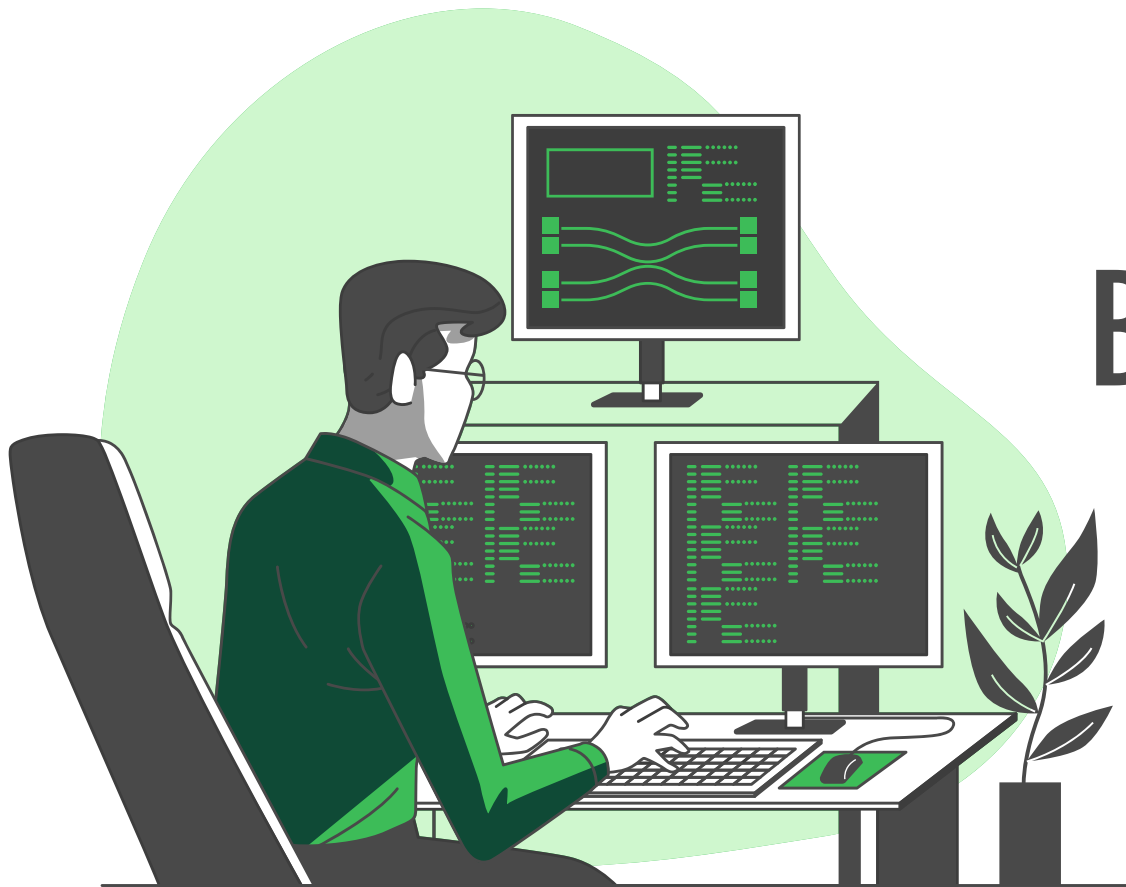
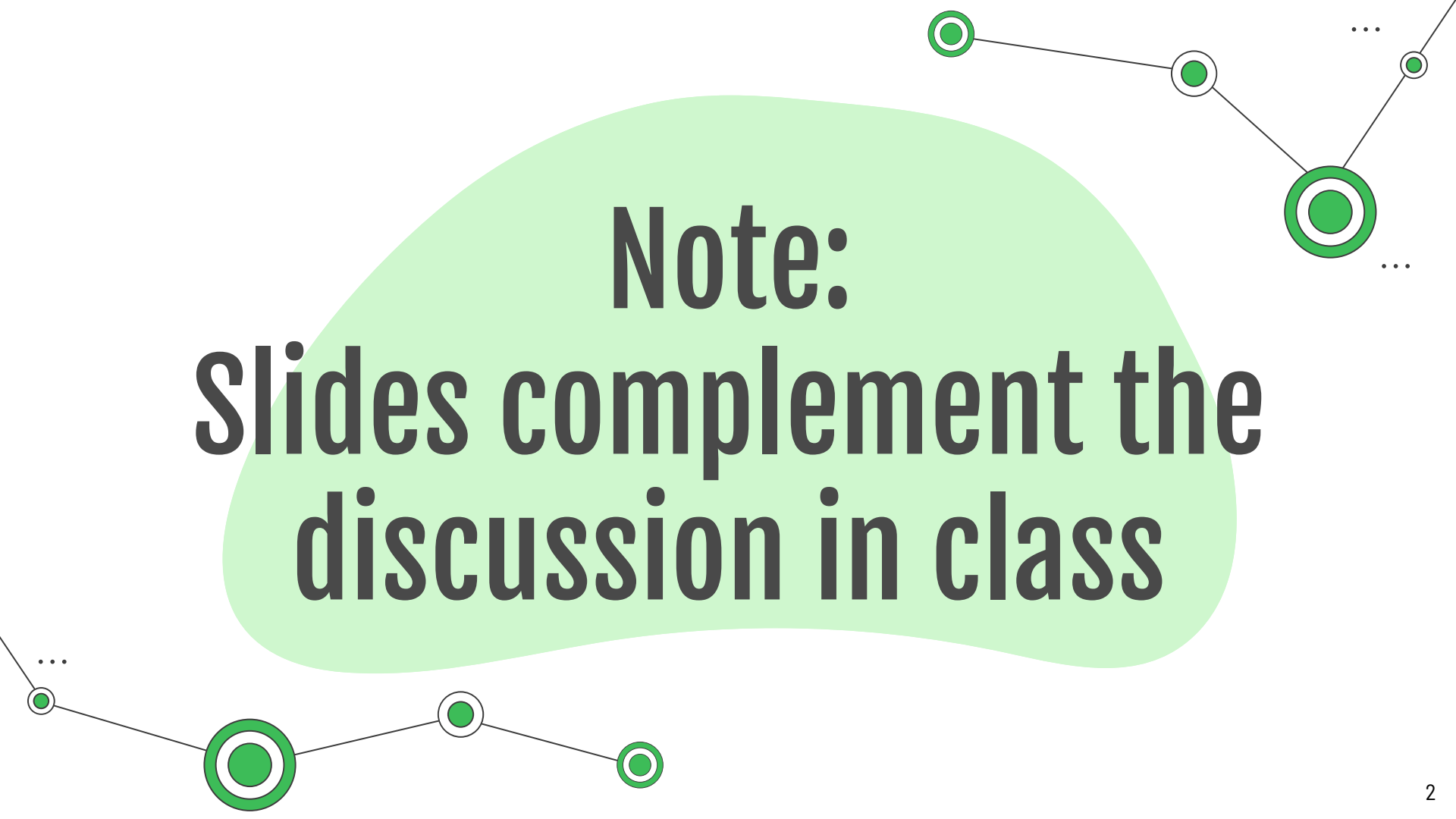




Bintree and K-D Tree

CS 251 - Data Structures
and Algorithms

A decorative network diagram consisting of several green circular nodes connected by thin black lines. Some nodes are single green circles, while others are double green circles. The nodes are arranged in a non-linear fashion, with some at the top right, some at the bottom left, and one in the center. Ellipses (...) are placed near some of the nodes, suggesting a larger, continuous network.

Note:
**Slides complement the
discussion in class**

Table of Contents

01

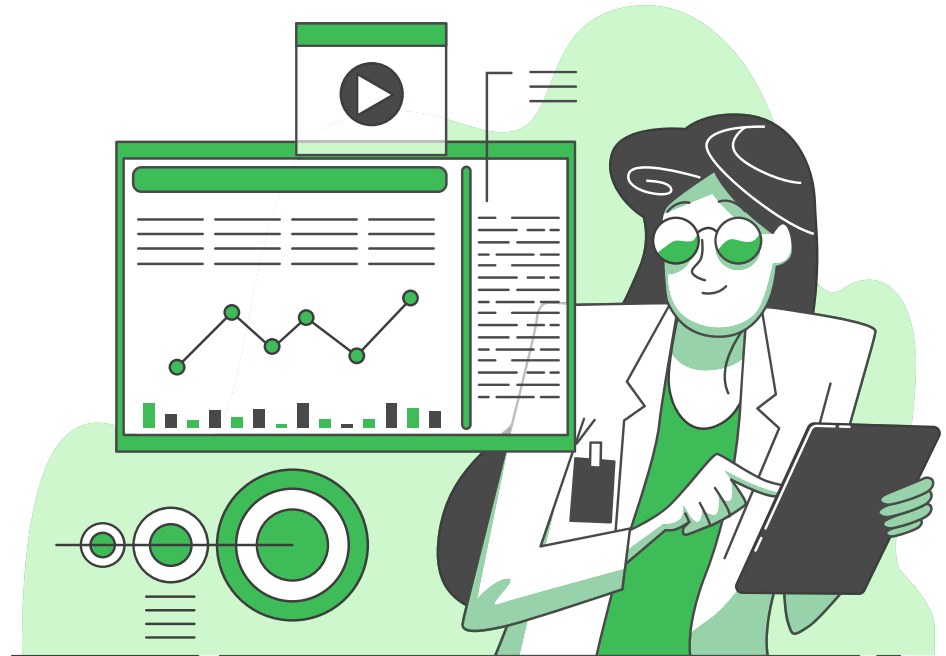
Bintree

Splitting multidimensional keys into equal halves

02

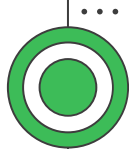
K-d Tree

Binary Search Trees for multidimensional keys



01 Bintree

Splitting multidimensional keys into
equal halves



...

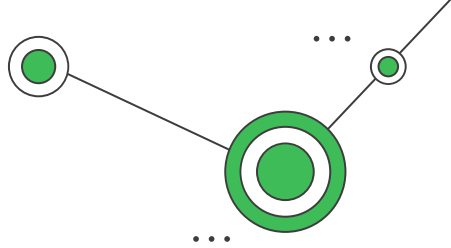


...



...

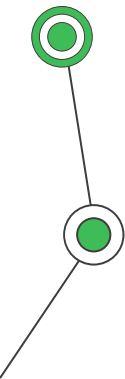
Bintrees



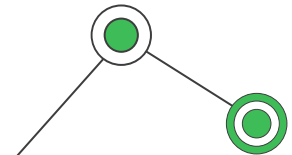
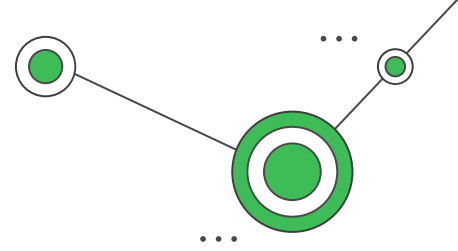
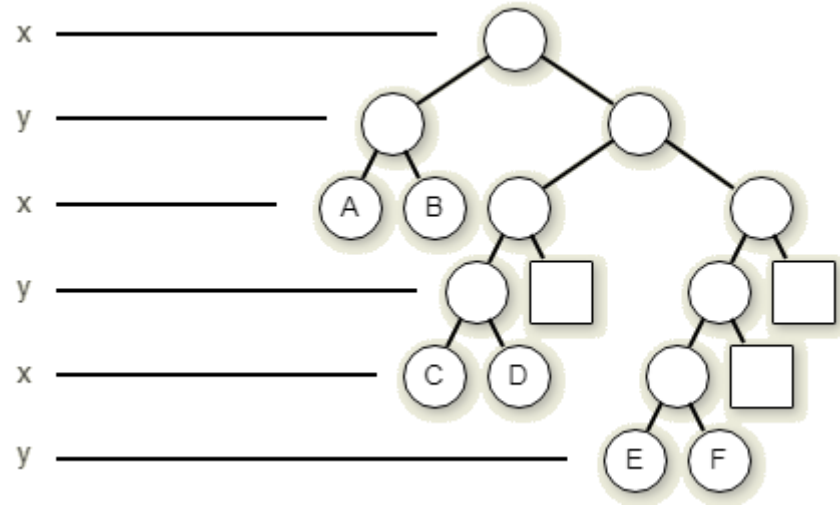
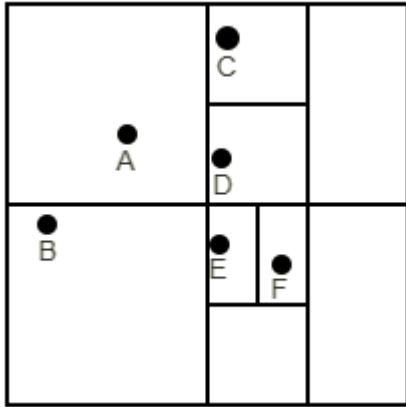
Think about it as BSTs for multiple keys.

That is, at each level, the branching happens with respect to a particular key associated with the level (aka. discriminator).

Split each space into equal halves rather than splitting at the key value.



Bintree Example





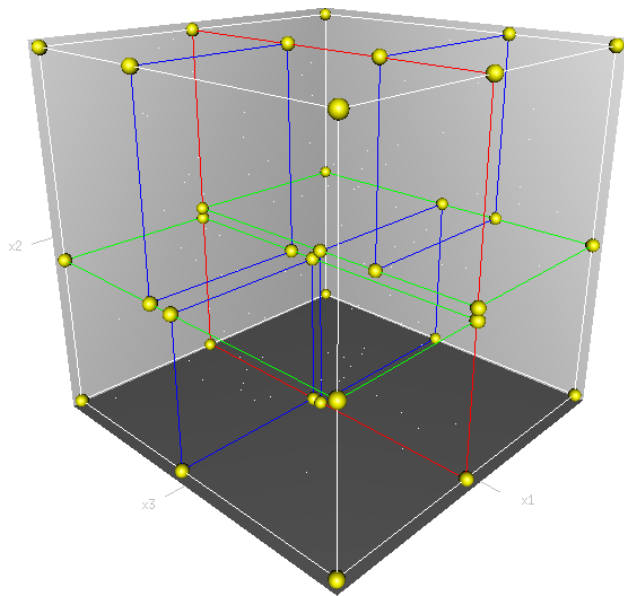
02

K-d Tree

Binary Search Trees for
multidimensional keys



K-d Trees



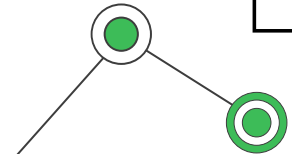
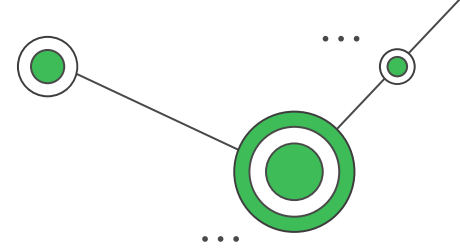
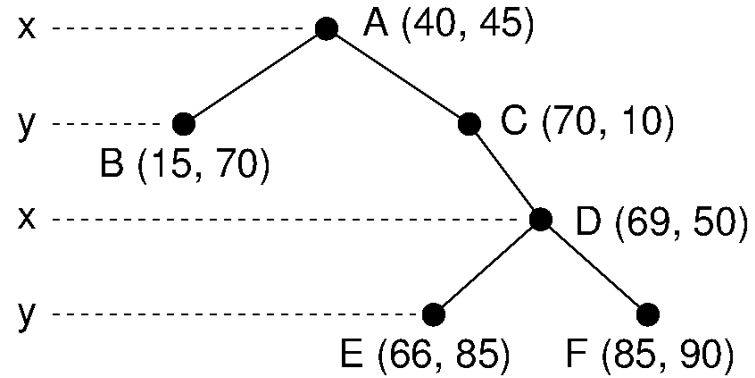
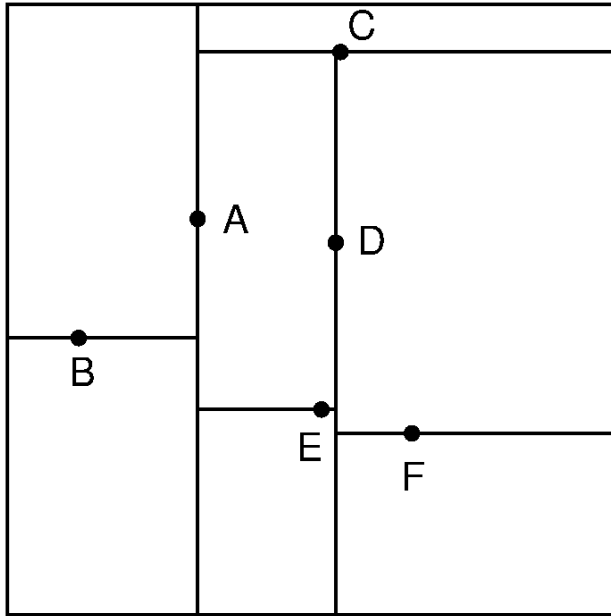
Think about it as BSTs for multiple keys.

That is, at each level, the branching happens with respect to a particular key associated with the level (aka. discriminator).

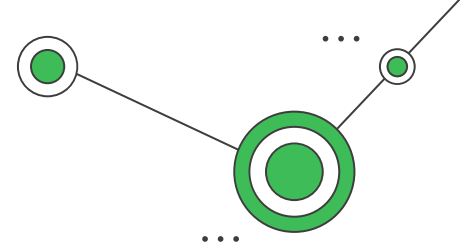
Every node is a k-dimensional point.

Every non-leaf splits the space into half-spaces.

K-d Tree Example

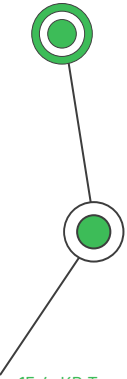


Single Search In K-d Trees

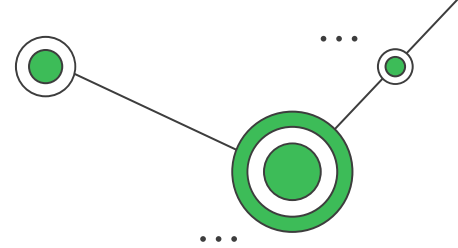


Let P be a point in a K-d Tree.

Searching a K-d Tree for the record with a specified xy-coordinate is like searching a BST, except that each level of the K-d tree is associated with a particular discriminator.

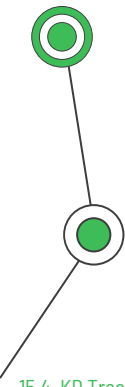


Region Search In K-d Trees

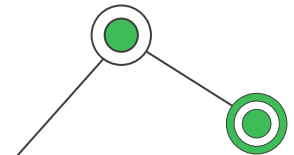
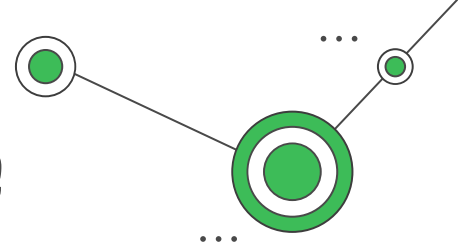
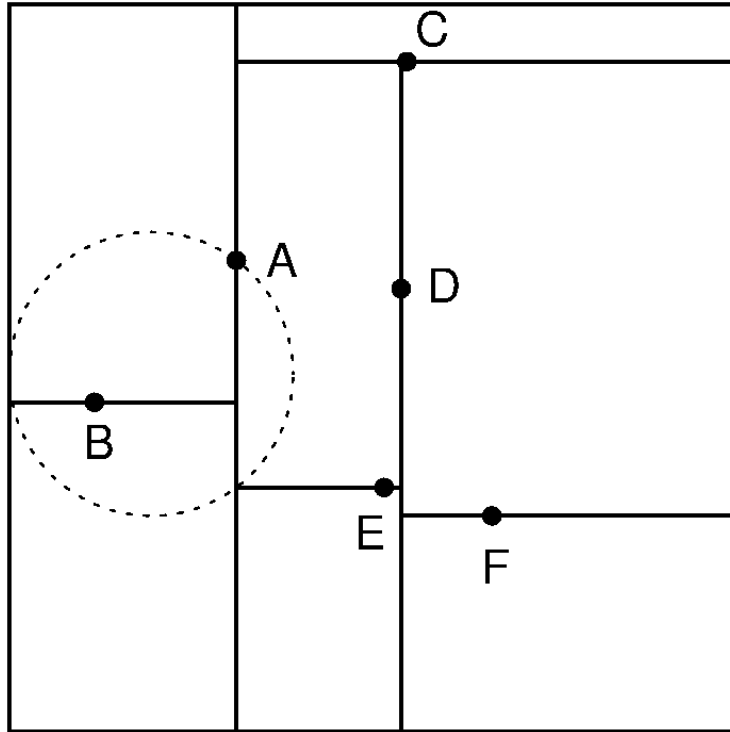


To locate all points within radius r of query point P :

- Use the Euclidean distance $d(P, N) = \sqrt{(P_x - N_x)^2 + (P_y - N_y)^2}$
- Reached a node with key value more than d above the value of the corresponding discriminator? Then no point in the respective right subtree is within range.
- Reached a node with key value less than d below the value of the corresponding discriminator? Then no point in the respective left subtree is within range.



Region Search in K-d Tree Example



Done!

Do you have any questions?

CREDITS: This presentation template was created by [Slidesgo](#), including icons by [Flaticon](#), infographics & images by [Freepik](#) and illustrations by [Stories](#)