Binary Search Trees and Rare Super Cars

By Cole P.

Definition

• A node-based data structure that compares keys in order to tell where a node goes



Ferrari P4/5

O Total Amount: 1

- Release Year: 2006
- Base Price: \$4,000,000

Info From:

<u>https://freeaddon.com/top-</u> <u>10-rarest-supercars-never-</u> <u>see/</u>



What does a node contain?

O Key

O Value

O Left Link

O Right Link



How it works

• The data structure takes the key from the data set and compares it to the "root" and continues from there



McLaren F1 LM

- Total Amount: 5
- Release Year: 1995
- Base Price: \$1,000,000

Info From:







Example (Insertion)



Pagani Zonda Revolucion

• Total Amount: 5

• Release Year: 2014

• Base Price: \$2,900,000

Info From:



Delete

- To delete a node you CAN find the successor of the node being deleted in the right tree and replace it
- BUT there are other methods for deleting nodes

Example (Deletion)

 In order to do this you would need to find the smallest value in the right branch of the node being deleted



Methods and Variables

Binary Search Tree

• Int Root

- Search(n)
- Insert(n)
- Minimum
- Maximum
- Delete(n)
- Floor
- Ceiling
- Selection
- Rank



- The search tree all runs in O(log n) because it depends on how many objects are in the structure
- Slower than O(1)
- Faster than O(n)

Different Cases



Lamborghini Veneno

- O Total Amount: 5
- Release Year: 2013
- Base Price: \$4,000,000

Info From:



Implementation

• Keep track of the root, the left, and right links then the rest of the tree is found from there



Applications of Search Trees

- Used in search applications
- 3D games for rendering



What's the Point?

Search trees CAN be more reliable than arrays and are faster than 0(n)
It is O(log n) meaning it must search, but not through all objects ideally

Koenigsegg One

O Total Amount: 6

• Release Year: 2015

• Base Price: \$2,400,000

Info From:





- O (log n)
- Adds items through simple method
- Less than Greater than operation

Quick and Reliable (not the fastest)