

Looking for a senior project?

Consider turning your first programming assignment (optimal binary search trees) into a manipulative tool to help other students learn the concepts.

We solved the Optimal Binary search tree three ways

(See <http://www.cs.usu.edu/~allanv/cs5050/cs5050.html> Problem 1)

- (1) Greedily
- (2) Using exhaustive recursion
- (3) Using Memoizing

A logical extension is to solve it via dynamic programming.

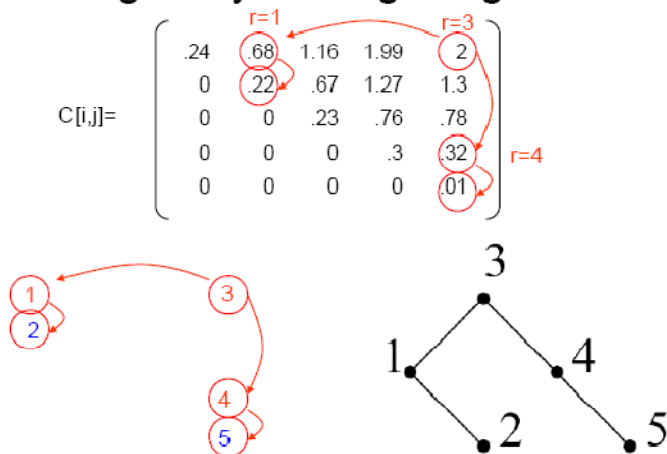
Things you could add:

1. Interact with the user so they can pick problem size or frequency values
2. Allow the user to pick a greedy strategy
 - a. Select largest frequency
 - b. Select node nearest the middle of the keys (to get a balanced tree)
 - c. Other strategies?
3. Show how you use dynamic programming to not only find the cost of the optimal binary search tree, but build it.

i	1	2	3	4	5
p_i	0.24	0.22	0.23	0.3	0.01

Backtrack the Optimal BST

- Recording every r during filling.



4. Show the tree you create graphically. See the demo at <http://webpages.ull.es/users/jriera/Docencia/AVL/AVL%20tree%20applet.htm> for ideas.

Perhaps you would show the nodes with a size proportional to its frequency.

5. Demonstrate the time required to use the various methods.