Programming Activity One

Represent a binary tree node as a structure with four components:

- key
- value
- left child
- right child

Programming Activity Two

Implement a method that returns a pointer to a free space suitable for holding a tree node.
struct tnode *talloc()

Binary search tree

- Invariant: for any node X, X’s key >= the keys in all nodes in X's left subtree; X’s Key < the keys in all nodes in X’s right subtree.
- In-order traversal: left subtree->root->right subtree

Programming Activity Three

In-order print of tree p
void treeprint(struct tnode *p)

Programming Activity Four

Node insert to a binary search tree
struct tnode *addtree(struct tnode * p, int k, int v)
Walk down the tree

1) if k is smaller than p's key, go to the left;
2) if k is larger than p's key, go to the right;
3) If k is equal to p's key, return.
4) when reach a null reference, replace that null reference with a new node, return.

i-clicker question

Which C library function frees the space created by malloc?

a) delete
b) remove
c) clean
d) free

i-clicker question

Which standard header are the malloc and free routine declared in?

a) stdlib.h
b) stdio.h
c) ctype.h
d) string.h