Binary Search Tree

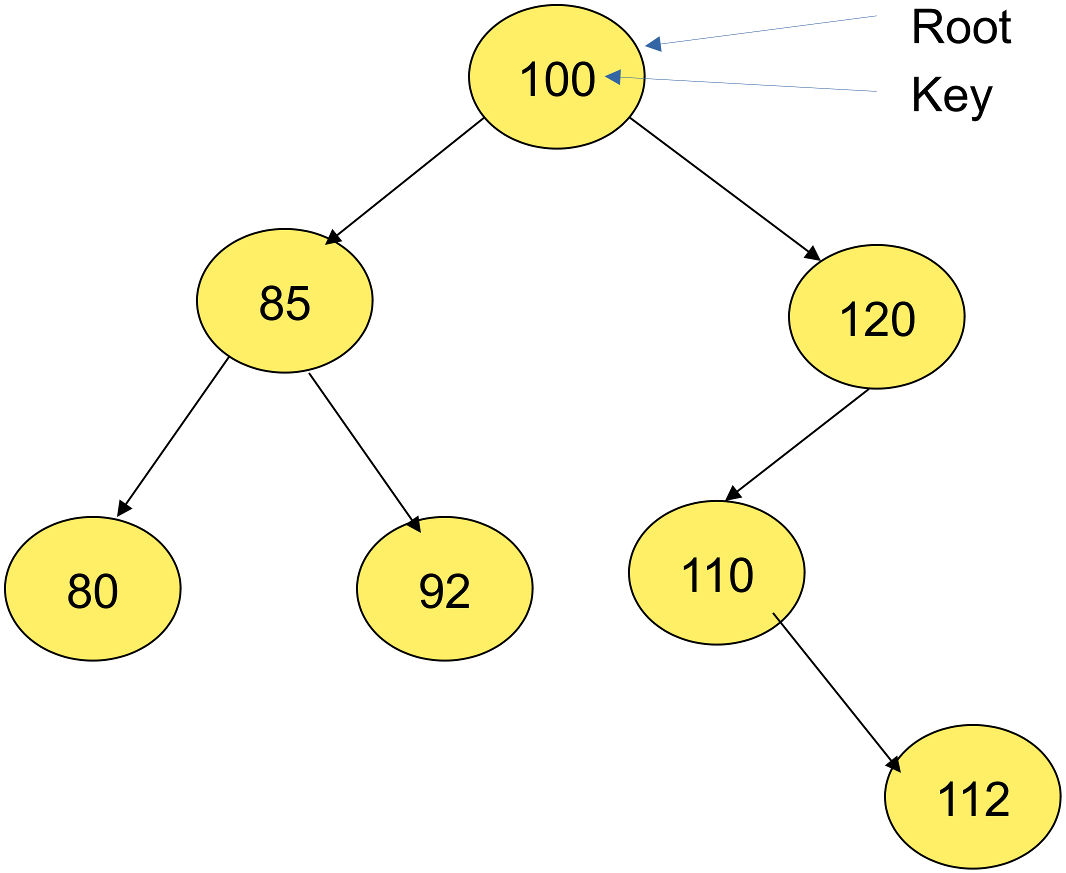
RM.Periakaruppan
Dept of AM&CS
PSG Tech

Binary Search Trees

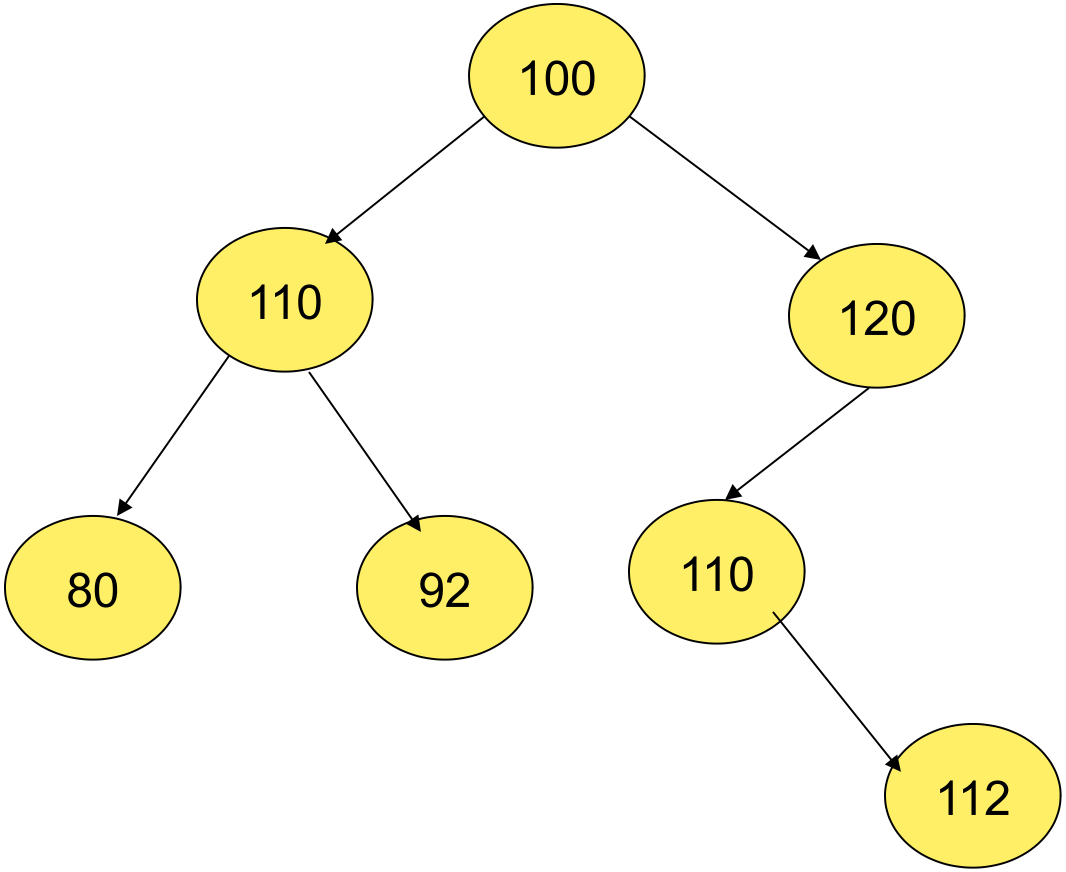
A binary search tree is a binary tree that may be empty. A non empty binary search tree satisfies the following properties:

- Every element has a key (or value), and all keys are distinct.
- The keys (if any) in the left subtree of the root are smaller than the key in the root.
- The keys (if any) in the right subtree of the root are larger than the key in the root.
- The left and right subtrees of the root are also binary search trees.

BINARY SEARCH TREE EXAMPLE



NOT A BINARY SEARCH TREE EXAMPLE

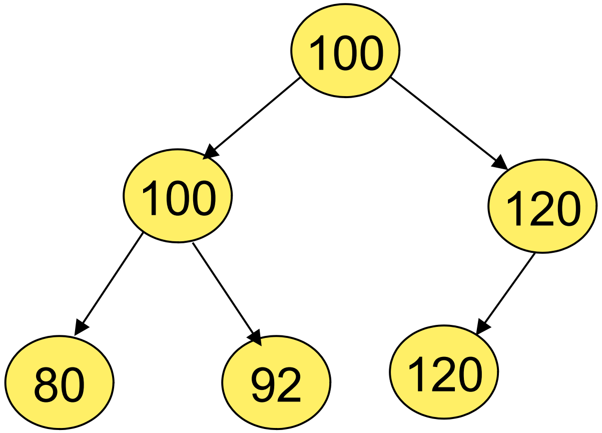


BINARY SEARCH TREE WITH DUPLICATES

A binary search tree in which all keys need not be distinct is called binary search tree with duplicates.

Here

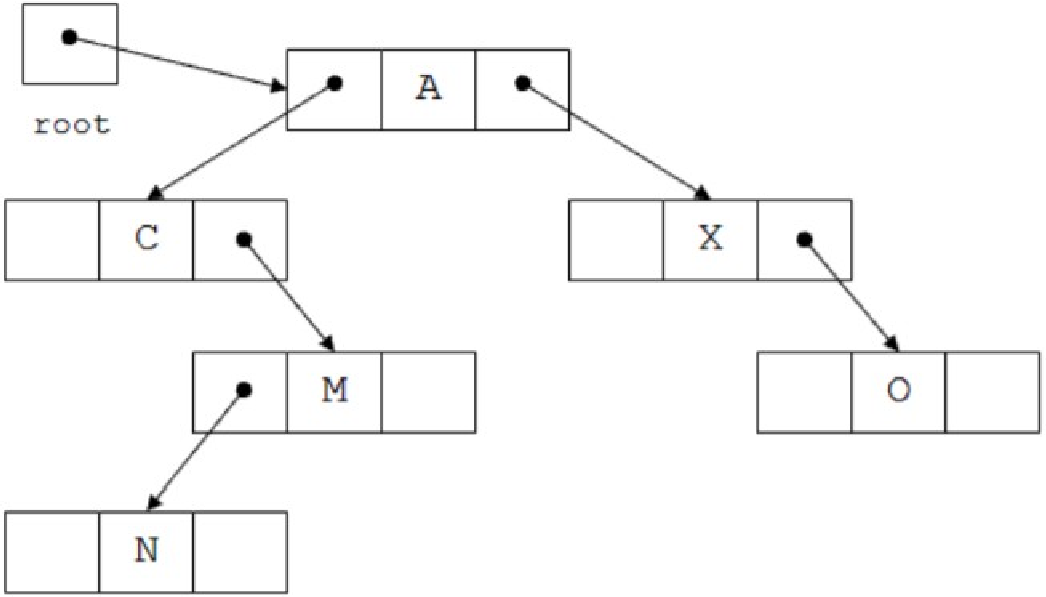
- The keys (if any) in the left subtree of the root are smaller than or equal to the key in the root.
- The keys (if any) in the right subtree of the root are larger than or equal to the key in the root.



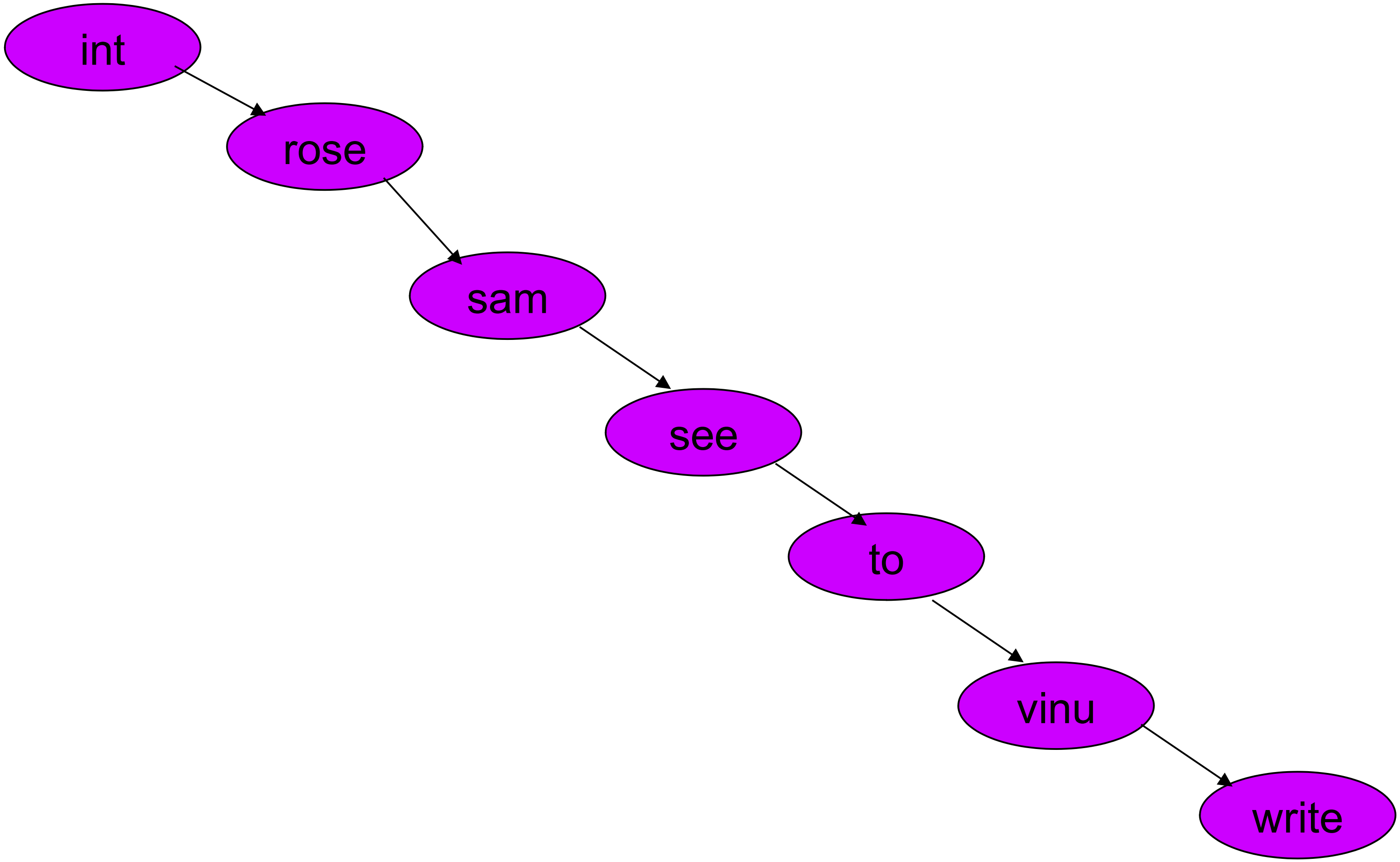
Structure of a node - C Representation

```
#define NUMNODES 500
struct nodetype {
  int info;
  int left;
  int right;
};
struct nodetype node[NUMNODES];
```

Linked Representation

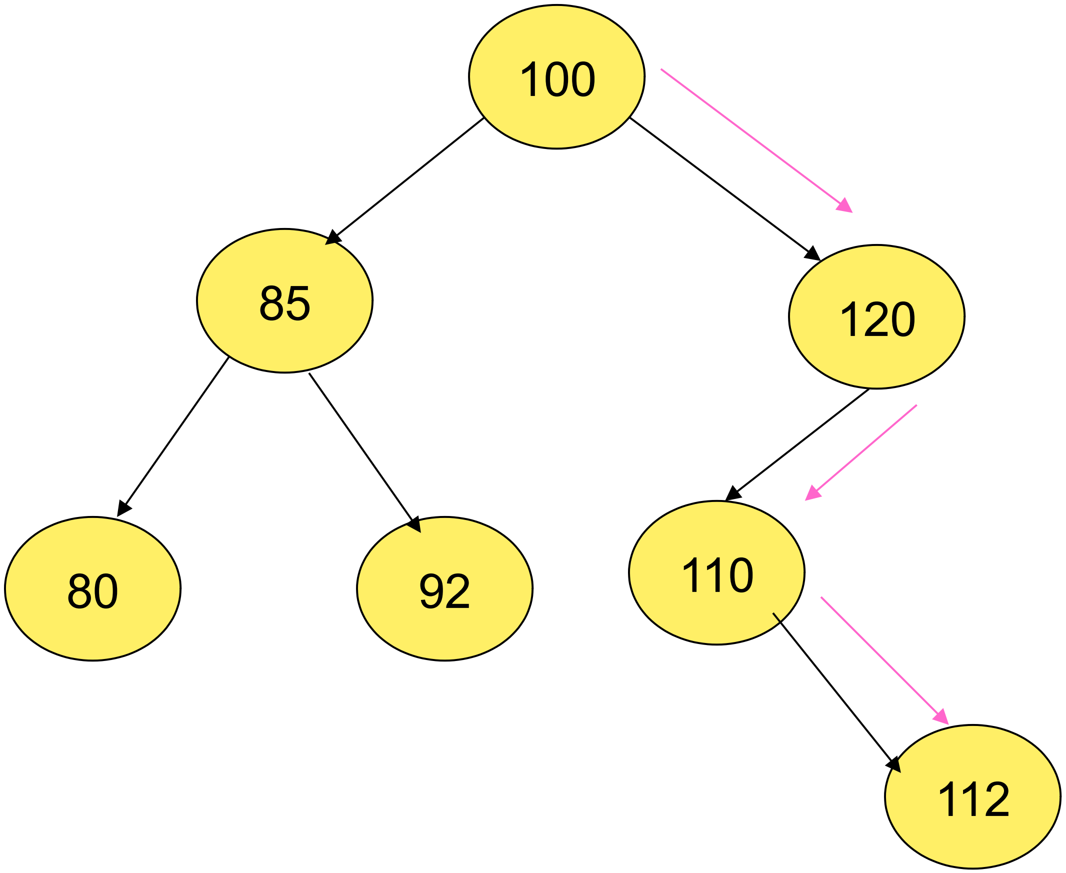


Skewed Binary Search Tree (Right)



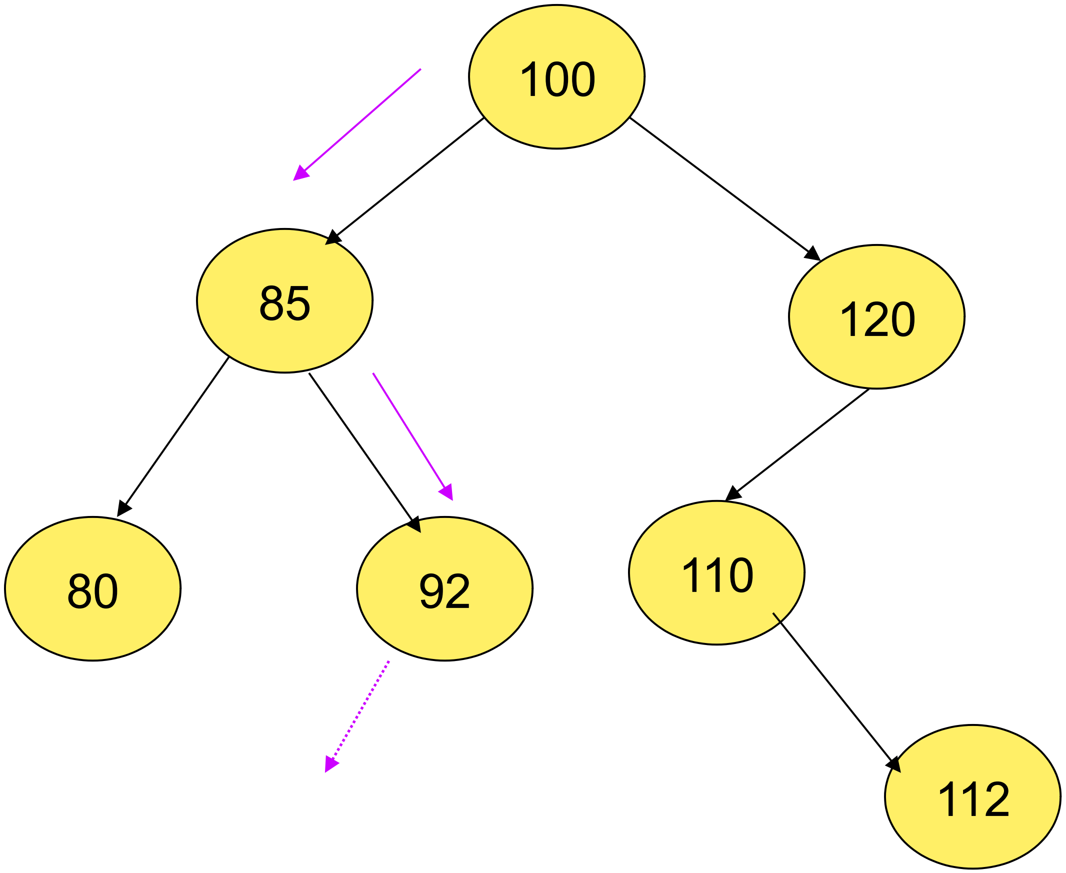
SEARCHING

SEARCH FOR 112



SUCCESSFUL SEARCH

SEARCH FOR 90



UNSUCCESSFUL SEARCH

Notations to be used in algorithm

p - Pointer to a node in tree

key(p) -   Key value stored in node pointed by p

right(p) - pointer to right node of p

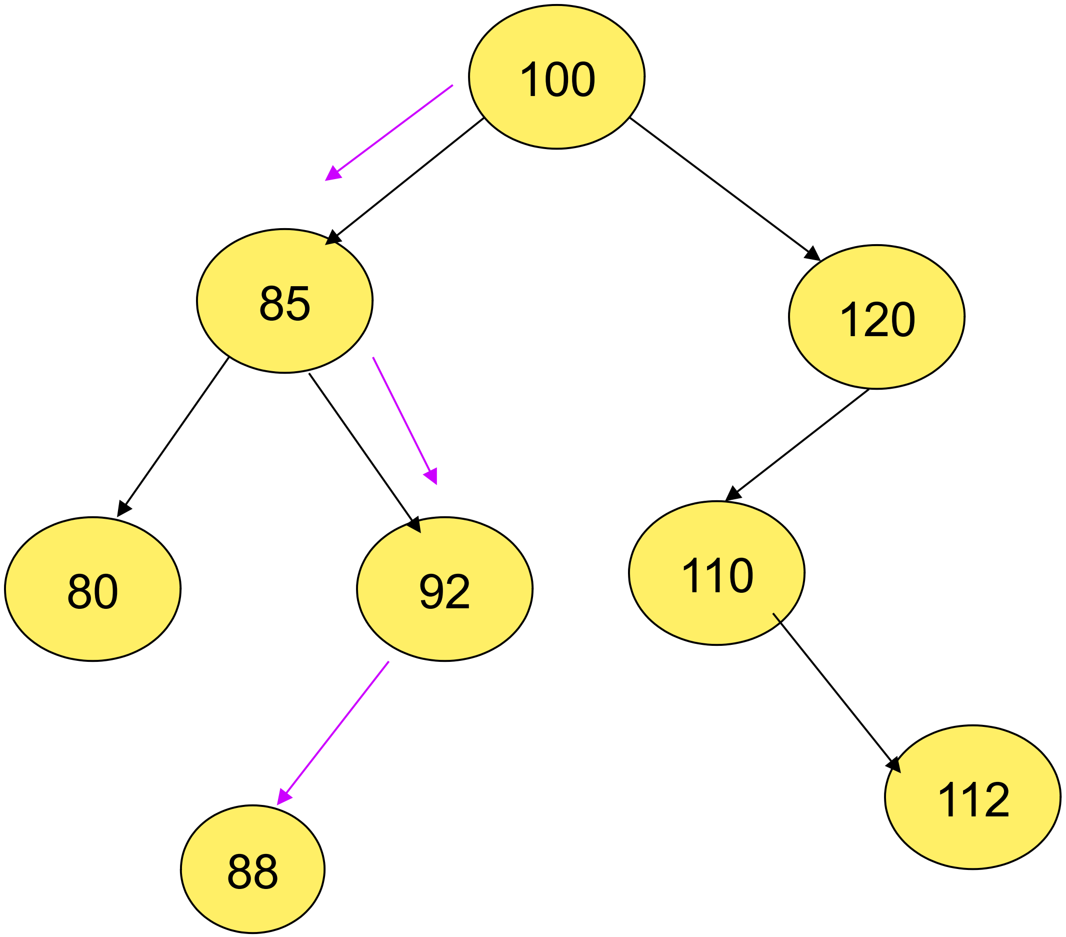
left(p)   - pointer to left node of p

SEARCHING algorithm

```
/* pointer p starts at the root and moves */
/* through the tree looking for an element */
/* with key k*/
p= tree
while (p != null)
{  If (k = key(p))
        return(p)
    else
        if (k < key(p))
            p = left(p)
        else
            p = right(p)
}
return (false)
```

INSERTION

INSERT 88



Questions

1. Suppose that we have numbers between 1 and 1000 in a binary search tree, and we want to search for the number 363. Which of the following sequences could not be the sequence of nodes examined?

a. 2, 252, 401, 398, 330, 344, 397, 363.

b. 924, 220, 911, 244, 898, 258, 362, 363.

c. 925, 202, 911, 240, 912, 245, 363.

d. 2, 399, 387, 219, 266, 382, 381, 278, 363.

e. 935, 278, 347, 621, 299, 392, 358, 363.

Questions

2. construct binary search trees for the sets

S1 = {A, B, C, D, E, F, G, H,I, J, K, L, M} and

S2 = {M, L, K, J, I, H, G, F, E, D, C, B, A}

Questions

3. For the set of 1; 4; 5; 10; 16; 17; 21 of keys,

draw binary search trees of heights 2, 3, 4, 5, and 6.