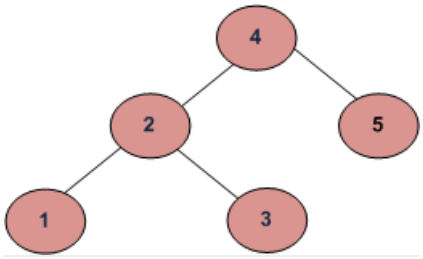
Sorted order printing of a given array that represents a BST

Given an array that stores a complete Binary Search Tree, write a function that efficiently prints the given array in ascending order.

For example, given an array [4, 2, 5, 1, 3], the function should print 1, 2, 3, 4, 5



Solution:

Inorder traversal of BST prints it in ascending order. The only trick is to modify recursion termination condition in standard Inorder Tree Traversal.

Implementation:

```
#include<stdio.h>

void printSorted(int arr[], int start, int end)
{
  if(start > end)
    return;

  // print left subtree
  printSorted(arr, start*2 + 1, end);

  // print root
  printf("%d  ", arr[start]);

  // print right subtree
  printSorted(arr, start*2 + 2, end);
}

int main()
{
  int arr[] = {4, 2, 5, 1, 3};
  int arr_size = sizeof(arr)/sizeof(int);
  printSorted(arr, 0, arr_size-1);
  getchar();
  return 0;
}
```

Time Complexity: O(n)