```
#include <iostream>
#include <stack>
using namespace std;

struct Node {
  int val;
  Node *left;
  Node *right;
};

void inorder(const Node *n, stack<int> &s) {
  if(!n) return;
  inorder(n->left, s);
  s.push(n->val);
  inorder(n->right, s);
}

bool check(const Node *root) {
  stack<int> s;
  inorder(root, s);
  int smaller, larger;
  while(s.size() > 1) {
    larger = s.top();
    s.pop();
    smaller = s.top();
    s.pop();
    if(larger < smaller) return false;
  }
  if(s.size() == 1) {
    int temp = s.top();
    s.pop();
    return smaller >= temp;
  } else {
    return true;
  }
}
```

```
#include <vector>
using namespace std;

struct Node {
  int val;
  Node *left;
  Node *right;
};

void inorder(const Node *n, vector<int> &arr) {
  if(!n) return;
  inorder(n->left, arr);
  arr.push_back(n->val);
  inorder(n->right, arr);
}

bool check(const Node *root) {
  vector<int> arr;
  inorder(root, arr);
  for(int i = 0; i < arr.size() - 1; ++i) {
    if(arr[i] > arr[i + 1]) return false;
  }
  return true;
}
```