vector<int>f(Node \* root){

if(!root) return vector<int>{0,1,INT\_MIN,INT\_MAX};

if(!root->left && !root->right) return vector<int>{1,1,root->data,root->data};

auto l=f(root->left);

auto r=f(root->right);

if(!l[1]||!r[1]||root->data<=l[2]||root->data>=r[3]) return {max(l[0],r[0]),0,0,0};

return {1+l[0]+r[0],1,max({l[2],r[2],root->data}),min({root->data,l[3],r[3]})};

}

int largestBst(Node \*root)

{

return f(root).front();

}