Vertical traversal of a binary tree

```
vector<int>
verticalOrder(Node *root)
                                    {
                                        map<int, map<int,vector<int> > > nodes;
                                        queue< pair<Node*, pair<int,int> > > q;
                                        vector<int> ans;

                                        if(root == NULL)
                                            return ans;

                                        q.push(make_pair(root, make_pair(0,0)));

                                        while(!q.empty()) {
                                            pair<Node*, pair<int,int> > temp =
                                    q.front();
                                            q.pop();
                                            Node* frontNode = temp.first;
                                            int hd = temp.second.first;
                                            int lvl = temp.second.second;

                                            nodes[hd][lvl].push_back(frontNode->data);

                                            if(frontNode->left)
                                                q.push(make_pair(frontNode->left,
                                    make_pair(hd-1, lvl+1) ));

                                            if(frontNode->right)
                                                q.push(make_pair(frontNode->right,
                                    make_pair(hd+1, lvl+1)));
                                        }

                                        for(auto i: nodes) {
```

```
            for(auto j:i.second) {


                for(auto k:j.second)
                {
                    ans.push_back(k);
                }
            }
        }
        return ans;
    }
};
```