/\* CAB ALLOTMENT PROBLEM \*/

#include <iostream>

#include <vector>

#include <algorithm> // Include the algorithm header for the sort function

using namespace std;

int main() {

vector<int> start;

vector<int> finish;

int n;

cout << "Enter the number of users: ";

cin >> n;

// Input start and finish times for each user

cout << "Enter start and finish times for each user:" << endl;

for (int i = 0; i < n; i++) {

int s, f;

cout << "User " << i + 1 << ": ";

cin >> s >> f;

start.push\_back(s);

finish.push\_back(f);

}

// Sort the intervals based on finish times

sort(start.begin(), start.end());

sort(finish.begin(), finish.end());

// Initialize variables

int cnt = 0; // Counter for cabs on road

int i = 0; // Index for finish times

int j = 0; // Index for start times

int cabno = 1; // Cab number

// Assign cabs to users based on their start and finish times

while (j < n) {

if (finish[i] <= start[j]) {

// If the finish time of the previous user is less than or equal to the start time of the current user,

// assign the current user to the same cab

i++;

j++;

cout << "User " << j << " in CAB " << cabno << endl;

} else {

// If there's a gap between the finish time of the previous user and the start time of the current user,

// assign the current user to a new cab

j++;

cabno++;

cnt++; // Increment the counter for cabs on road

cout << "User " << j << " in CAB " << cabno << endl;

}

}

// Output the total number of cabs on road

cout << "Total number of cabs on road: " << cnt << endl;

return 0;

}