**SOLUTION**

class Solution {

public:

Solution(){

ios::sync\_with\_stdio(false);

std::cin.tie(nullptr);

std::cout.tie(nullptr);

}

vector<vector<int>> intervalIntersection(vector<vector<int>>& A, vector<vector<int>>& B) {

if(A.size() == 0 || B.size() == 0)

return {};

vector<vector<int>> result;

int i=0,j=0;

int m=A.size(), n=B.size();

while(i<m && j<n){

int left = max(A[i][0],B[j][0]);

int right = min(A[i][1],B[j][1]);

if(left<=right)

result.push\_back({left,right});

if(A[i][1]<B[j][1])

i++;

else

j++; }

return result; }

};

**TIME COMPLEXITY= O(M+N) || O(N)**

**SPACE COMPLEXITY= O(N)**