<https://www.hackerrank.com/challenges/beautiful-path/problem>

let edges = [

[1,2,1],

[1,2,1000],

[2,3,3],

[1,3,100]

] ;

let A = 1;

let B = 3;

let firstOccurence = null;

let destOccurence = null;

let M = 3;

let n = 4;

//first occurence of A

edges.forEach((val,ind)=>{

for(let k=0; k<M; k++){

if(firstOccurence === null && val[k]===A){

firstOccurence = {

i : ind,

j : k

}

}

}

for(let k=0; k<M; k++){

if(destOccurence === null && val[k]===B){

destOccurence = {

i : ind,

j : k

}

}

}

});

let i = firstOccurence.i;

let j = firstOccurence.j;

let bestpath = 0;

looper = ((i,j)=>{

console.log("find",edges[i]);

if(i<n && j<M){

let bottom = edges[i+1] && edges[i+1][j] ? edges[i+1][j] : "x";

let right = edges[i] && edges[i][j+1] ? edges[i][j+1] : "y";

if(bottom==="x"){

bottom = right+1;

} else if(right==="y") {

right = bottom+1;

} else if(bottom==="x" && right==="y"){

return -1;

} else {

console.log("pass");

}

let current = {

index : null, value : null

}

if(bottom === B || right === B) {

return B;

} else if(bottom>right) {

current.index = {i:i,j:j+1};

current.value = right;

} else if(bottom<right) {

current.index = {i:i+1,j:j};

current.value = bottom;

} else if(bottom===right) {

// if(edges[i+2] && (edges[i+2][j] < edges[i+1][j+1])){

// current.index = {i:i+1,j:j};

// current.value = bottom;

// } else if (edges[j+2] && (edges[i][j+2] < edges[i+1][j+1])) {

// current.index = {i:i,j:j+1};

// current.value = right;

// } else if(!edges[i+2] && edges[j+2]){

// current.index = {i:i,j:j+1};

// current.value = right;

// } else if(edges[i+2] && !edges[j+2]){

// current.index = {i:i+1,j:j};

// current.value = bottom;

// }else if(!edges[i+2] && !edges[j+2]){

// current.index = {i:i+1,j:j};

// current.value = bottom;

// } else {

// current.index = {i:i+1,j:j};

// current.value = bottom;

// }

current.index = {i:i+1,j:j};

current.value = bottom;

} else {

return -1;

}

bestpath = bestpath+current.value;

looper(current.index.i,current.index.j);

} else {

return -1;

}

});

looper(firstOccurence.i,firstOccurence.j);