Maximum Bipartite Matching

Si el matching no es maximo, existe un augmenting path del grafo (camino alternante entre aristas del match y no del match, el inicio y el final del camino son vertices no en el match)

Asi que mientras encuentres match, puedes hacer el match mas grande

Lista adj,

Si yo estoy en match, y mi vecino tambien, la arista es match

Si yo no estoy en match o mi vecino no, la arista no es match

Con dfs digo si estás en un conjunto o en el otro

Bool enA[] o enIzq[]

Via Augmenting paths:

int match[maxN] .//te dice con quien estas asociado

* BFS:

Arreglo papas[

Para cada nodo en A, si eres free, te meto a BFS

En BFS: while no vacia && no final

Pop();

If(visited) continue;

Visited[] = true

Papa[nd] = papa

Si soy de B,

Si soy free, me marco como final, break;

Si match, meto a Bfs a mis hijos de A unidos con aristas de match

Si soy de A, (soy match)

Meto a bfs a mis hijos de B unidos con aristas free

Cuando salgo, if vacia, el matching es maximo

Else

Vacio el arreglo match

While(yo no sea mi propio papa

Si soy de b, asocio a mi papa conmigo y viceversa

Final = papa;

Si soy de a, final = papa;

Repito

Codeforces round 92

Si tienes que hacer decisiones, checa si puedes fijar las decisiones e iterar sobre cuantos sí /no haces.

Si hay dos tipos de eventos, y un evento excluye a algunos del otro tipo, puedes obtener el maximo que coexisten obteniendo el max Independent Set = vertices – tamanio max Bipartite matching

UH LA LA CHULADA.

Idea del MST

Prim, kruskal

Teoream 11.1? del algorithms

Si no estás en el óptimo, construyo uno igual óptimo contigo

Te uno, en el simple path se forma un ciclo, elijo cualquiera d elos del arbol, lo corto y se separa en dos componentes y contigo las uno.