Problem Statement:

->Given N cards and every cards have two faces which consist 2 numbers, we have to put every card on the table such that maximum number of distinct characters present on the table.

->for every ith card it has two sides with number ai and bi

->if we transform the problem in graph , and consider these two sides is two nodes connected by and edge

->then our problem will become find maximum number of incoming edges toward every node

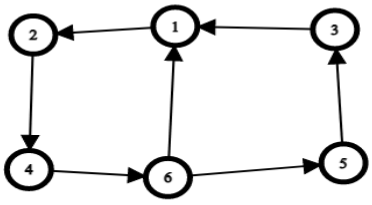
Observation;

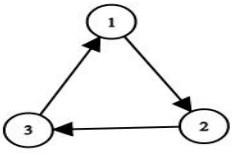
->when we first consider them as a directed graph, there will be three case occur

->(i) graph can be tree means it has (n-1) edges

->(ii) graph consist a cycle with means every node have atleast one inward edges

->we will try Union find so that we can find there is a cycle or not





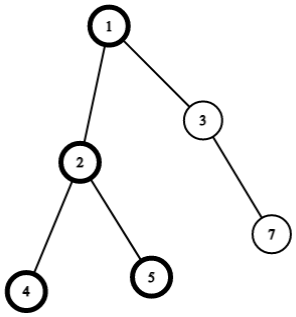


Fig:1 Fig:2 Fig:3

In fig:3 ,we can see there is only (n-1) edges , so maximum incoming edges we can make is (n-1) ,and there is no we can make it better so answer will be n.

Here if edges (1->2) means if card has two sides with number 1 and 2 then

If incoming edges towards two means in the final table we show that face of the cards which consist number 2.

In fig:2 because there is cycle we can make n incoming edges , which is maximum answer.

In fig:1 Actually we have cycle but, there is (n+1) incoming edges but our answer maximum can be n ,we have to take care of that we don't overcount the answer;

Implementation:

->we keep track of visited array if vis[u] is true it implies that we are already in cycle and there is one incoming edge towards you to avoid over count (in Fig:1)

->By using Union find we check if the graph is tree is tree then answer will (n-1)fig:3

->but if graph has cycle then answer will be N