Solution for the Team Builder Game
Joseph Qin 11/12/2017

Problem

This problem can be simply described as:
In a directed graph, to find out **how many nodes can be reached by all other nodes** and **how many nodes can reach all other nodes**.

Solution

take the one of the given inputs for example:
{"0010","1000","1100","1000"}
actually describes a directed graph as left picture shows.

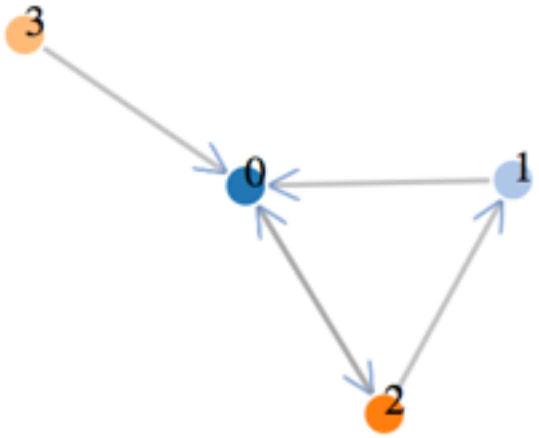
In Javascript we can store this graph as a nest list
the index is the node ID and
the values means all the nodes which can be reached this node by one step:

the left graph can be stored as:
var **graph = [[2], [0],[0,1],[0]]**

graph[i] = [the list of nodes which can be reached by node i by one step]

if graph[i] = [empty], then means node i can not reach any other node by one step

In this scenario:
we don't care about the paths or steps between two nodes.
in other words, if node 3 can go through node 0 and then reach node 2, we also consider node 2 is reachable by node 3.



so for each node, we need find all descendants of it.
(same as treat each node as root node and set up tree structure of it and then find all descendants)

this graph can be described as such four lists:
0->2->1
1->0->2
2->1->0
3->0->2->1

Then we just need check

1) how many paths contains all nodes which equals how many nodes can reach all others
2) how many common nodes among all paths which equals how many nodes can be reached by all others

Coding:

step1: parseGraph(): convert input string list to two dimension array

step2: travers each node by getAllReachabledNodes() to get a reachable list of each node

step3: find how many reachable list contains all nodes and how many common nodes among all reachable lists

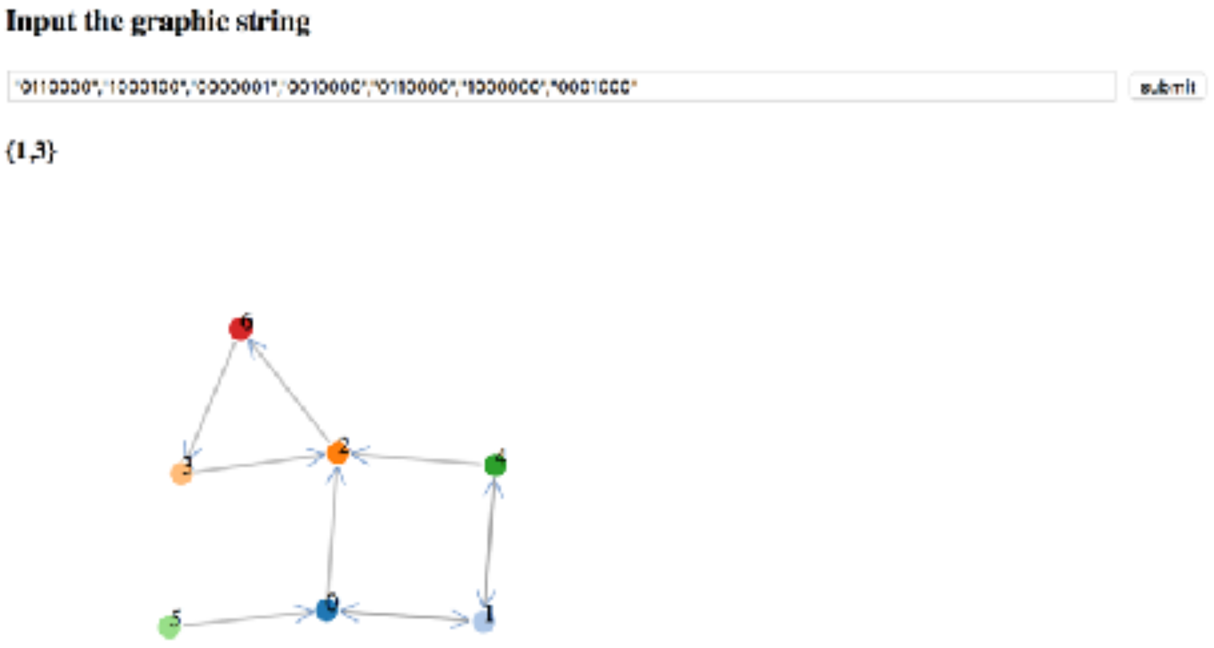
Files:

teamBuilder.js: Class teamBuilder and all functions

teamBuilder.html:

Web API for demonstration my resolutions of the team builder problem.

Running environment:
1) Open this html in Google Chrome/Firefox/Safari
2) Input the graph string in the input box and then click submit
3) it will render the graph layout, while users also can customize the layout by dragging
4) the special numbers will be showed between input box and graph layout



Input the graphic string

"0110000","1000100","0000001","0010000","0110000","1000000","0001000"

{1,3}