→ Is conned islands ⇆ Word ladder

→ Bipartite ⎤
→ K distace ⎦ ☆ ☆

→ Bell Man fords

Cb.lk,/ leet code

put: grid = [
["1","1","1","1","0"],
600 × 330
["1","1","0","1","0"],
["1","1","0","0","0"],
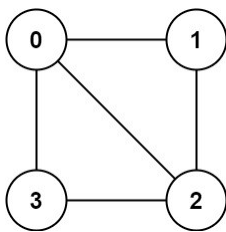["0","0","0","0","0"]

From <https://leetcode.com/problems/number-of-islands/>

[["1","1","0","0","0"],
["1","1","0","0","0"],
["0","0","1","0","0"],
["0","0","0","1","1"]

Input: beginWord = "hit", endWord = "cog", wordList =
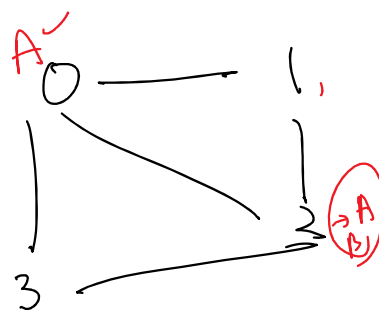["hot","dot","dog","lot","log","cog"]

From <https://leetcode.com/problems/word-ladder/>

graph = [[1,2,3],[0,2],[0,1,3],[0,2]]

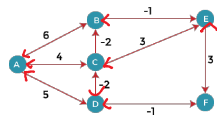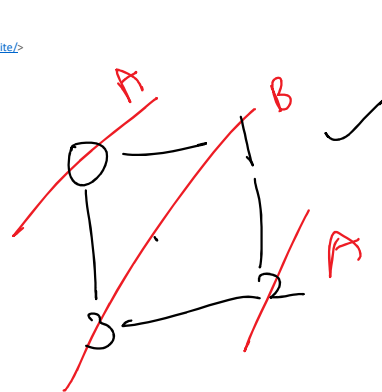From <https://leetcode.com/problems/is-graph-bipartite/>

$0 \to 1, 2, 3$
$1 \to 0, 2$
$2 \to 0, 1, 3$
$3 \to 0, 2$

Lets :

A

$0 \to A$
$1 \to B$
$2 \to B$
$3 \to B$

$\to A$
$\to B$

0 | 1 | 2 | 3

graph = [[1,3],[0,2],[1,3],[0,2]]

$V_1 \neq V_2$

A B

A F
V-1

A,A,1² · Sig · X

in of cost
Hot set

A —2— B   A ,   C, 2
3   4   A,C,1
   C   A,A,1 + 3
               A,B,1+4
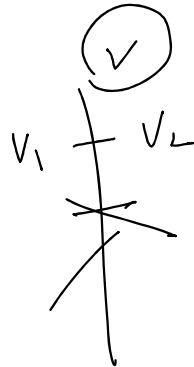
```java
private int BellMan(int scr, int dest, int cnt) {

        if(scr == dest) {
                return 0;
        }
        if(cnt==0) {
                return Integer.MAX_VALUE/2;
        }
        int sp = BellMan(scr, dest, cnt-1);
        for(int nbr : map.get(dest).keySet()) {
                int path_cost = BellMan(scr, nbr, cnt-1)+getEdge(nbr, dest);
                sp = Math.min(path_cost, sp);
        }
        return sp;
    }
```