Homework 9: Q2

Name: Xinkai Lin, xinkaili

*Don't forget to input your list of collaborators and sources on **AutoLab**.*
Please submit this file as a PDF.

1 Algorithm Idea

According to the problem, giving input a directed graph G = (V, E), a vertex $s \in V$ is called a sink. By definition, at s row of the adjacency-matrix will be all 0, and the s column will be all 1 except on s row. Starts from first row going from left to right column-wide. i represent row and j represent column. if $A_{ij} = 1$, then i = j, i++, go to next column but in j row. When this loop end, I will find the potential s vertex. Now, to check if s is sink. For $1 \leq k \leq n$, if $A_{sk} = 1$, this means there exit 1 in s row which violate the property of sink, so return null. If $A_{ks} = 0$ and s not equal k, this means at s column there exits 0 other than s row s column, which also violate the property of sink, so return null. If the above two cases do not exit, return s.

2 Algorithm Details

Input: G in its adjacency matrix, n the size of vertex

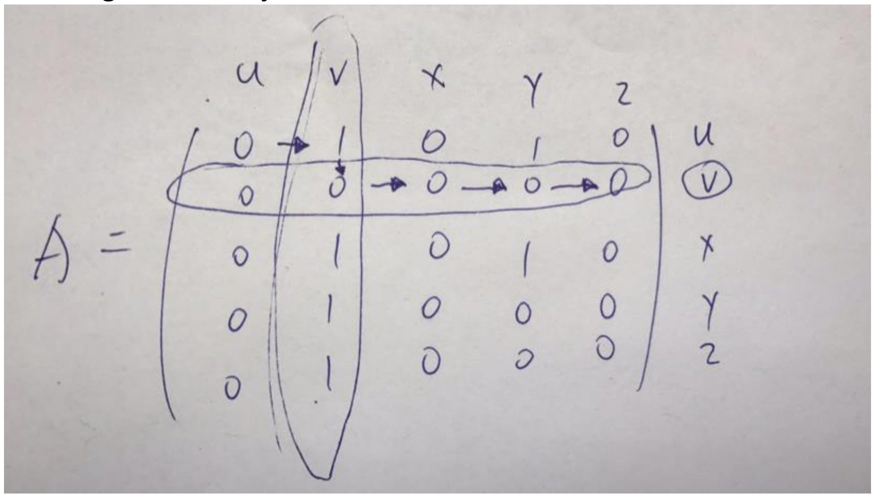
```
s=0
for (i=0; i<n; i++) do
    if (A[s][i] == 1) do
        s = i
    end if
end for
for (j=0; j<n; j++) do
    if (A[s][j] == 1)   return null
    if (A[j][s] == 0 && s!=j)   return null
end for
return s
```

3 Proof of Correctness Idea

To prove the algorithm's correctness, we will find the s vertex which qualified a sink's property. In this algorithm, first it will iterate all the vertices from left to right, if the entry is 0, keep going right until the last vertex. If the entry is 1, keep going right but jump to the s row where has encounter 1. When this loop end, it will find the potential s vertex that may be a sink. Then on the second for loop, it will check for the property of sink to ensure the output. First it will check at s row, if there exits 1, then it can't be a sink, return null. Then, it will check at s column, if there exits 0 except at s row s column, then it can't be a sink as well, return null. Otherwise, s is a sink and return s.

4 Proof Details

Starts from A_{11}, going from left to right. If $A_{ij} = 0$, then keep going right, else if $A_{ij} = 1$, then i=j, meaning i now is at j row in this case is v. Until it hit the last vertex.



Then, it will check for the property where at row v all the entry is 0, at column v, all the entry is 1 except A_{vv}.

5 Runtime Analysis

On the first for loop, it will iterate all the vertices from left to right which has size of n. Inside the for loop each entry will take O(1) time, so this make it O(n) time. On the second for loop, it will once again iterate all the vertices, each conditional statement will take O(1) to determine whether it qualified the property of a sink. This will also take O(n) time. So the overall runtime for this algorithm is O(n) + O(n), which is O(n) time.