Homework08 2021.12.4

1

给定 $G=(V,E)$ 是一带权重且没有权重为负值的环路的有向图，对于所有的结点 $v\in V$，从源结点s到结点 v之间的最短路径中，包含边的条数的最大值为 m。请对算法 $BELLMAN-FORD$ 进行简单修改，可以让其在 $m+1$遍松弛操作之后终止，即使 m不是事先知道的一个数值。

由于在任意一轮循环中对许多边的松弛不会成功，只有在上一轮中d值发生变化的顶点指出的边才有可能松弛成功，改变其他点的d值。所以当在某次循环结束时的d值与该次循环之前的d值相同时，结束外层循环。

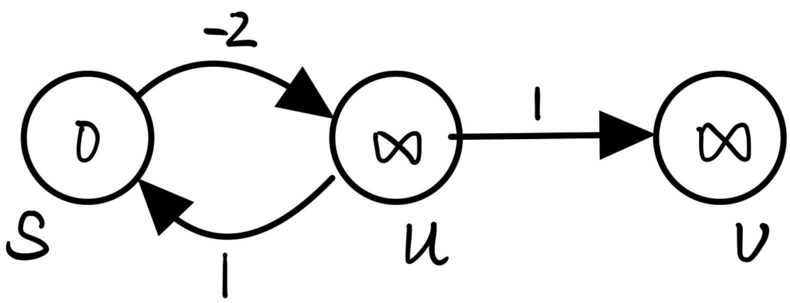
```
BELLMAN-FORD(G,w,s)
INITALIZE-SINGLE_SOURCE(G,s)
flag = 1
for i = 1 to |G.v|-1
    if flag == 1
        for each v in G.v
            save v.d to array
        for each edge(u,v) in G.E
            RELAX(u,v,w)
        for each v in G.v
            if v.d != array[].d
                flag = 1
                break
            else
                flag = 0
    else
        break

for each edge(u,v) in G.E
    if(v.d > u.d + w(u,v))
        return FALSE
return TREU
```

2

请举出一个包含负权重的有向图，使得 $Dijkstra$ 算法在其上运行时将产生不正确的结果。为什么在有负权重的情况下，这一定理的证明不成立？

当图中包含有权值为负值的环路时，会在环路上无限循环，算法无法输出正确的结果。如下图所示；



当存在权重为负值时，就无法在证明的过程中保证$\delta(s,y) \leq \delta(s,u)$始终成立。

3

$Floyd-Warshall$算法的空间需求为$\Theta(n^3)$，因为要计算$d_{ij}^{(k)}$，其中$i,j,k=1,2,...,n$。请证明下面所列出的去掉所有上标的算法是正确的，从而将 $Floyd-Warshall$算法的空间需求降低到$\Theta(n^2)$。

```
Floyd-Warshall'(W)
1: n = W.rows
2: D = W
3: for k = 1 to n do
4:      for i = 1 to n do
5:          for j = 1 to n do
6:              dij = min(dij , dik + dkj)
7: return D
```

原$Floyd-Warshall$算法中上标的作用是区分不同次最外层循环的结果，而去掉上标，使用一个矩阵保存结果，需要考虑覆盖是否会对算法的正确性造成影响。

考虑原$Floyd-Warshall$算法中的核心语句：$d_{ij}^{(k)}=min(d_{ij}^{(k-1)},d_{ik}^{(k-1)}+d_{kj}^{(k-1)})$，其含义是取上一次循环中$d_{ij}$与$d_{ik}+d_{kj}$的最小值。

考虑去掉上标的$Floyd-Warshall$算法，在计算某个d_{ij}时，此时的d_{ij}还是上一次循环的d_{ij}，d_{ij}不会被覆盖；

考虑后面的加法项，d_{ik}和d_{kj}可能被覆盖，若没有被覆盖，则去掉上标对于$Floyd-Warshall$的正确性没有影响；假设d_{ik}被覆盖，则d_{ik}的实际意义为$d_{ik}^{(k)}$，即从结点i到结点k的所有中间结点全部取自集合$\{1,2,...,k\}$的一条最短路径的权重，与d_{kj}（d_{ij}可能也被覆盖，若被覆盖，与d_{ik}同理）相加后得到从i到j的路径的所有中间结点还是全部取自集合$\{1,2,...,k\}$，因此并不会破坏$d_{ij}^{(k)}=min(d_{ij}^{(k-1)},d_{ik}^{(k-1)}+d_{kj}^{(k-1)})$的含义，以及$Floyd-Warshall$算法的核心思想。因此$Floyd-Warshall$算法的正确性仍然成立。