第 *3-8* 讲*:* 单源最短路

姓名： 赵超懿　　**学号：** *191870271*

评分： ______　　**评阅：** ______

2020 年 *11* 月 *7* 日

请独立完成作业，不得抄袭。 若得到他人帮助, 请致谢。 若参考了其它资料，请给出引用。 鼓励讨论，但需独立书写解题过程。

1　作业（必做部分）

题目 1 (TC 24.1-2)
Prove Corollary 24.3.

解答:
充分性:
当存在一条节点 s 到节点 v 的路径时，将路径上的边相加即可得到 s 到 v 的最短路径的一个上界 W，而算法终止时,$v.d = \sigma s, v < W < \infty$
必要性:
当 $v.d < \infty$ 时，沿着算法得到的结果有了 s 到 v 的路径（即为递归的打印当前节点的 $v_i.\pi$）, 这样得到的路径是 s 到 v 的一条路径。

题目 2 (TC 24.1-3)
Given a weighted, directed graph G=(V,E) with no negative-weight cycles,let m be the maximum over all vertices $v \in V$ of the minimum number of edges in a shortest path from the source s to v. (Here, the shortest path is by weight, not the number of edges.) Suggest a simple change to the Bellman-Ford algorithm that allows it to terminate in m+1 passes, even if m is not known in advance.

解答:
将对每个边进行松弛的 for 循环改为 while 循环，循环条件为当每次循环时有边被松弛，当某次循环没有边被松弛时，循环结束。
包含边数最多的边最短路径为 m, 故 m 次循环后之后的循环不会再有边被松弛，故循环在第 m+1 次停止。

题目 3 (TC 24.1-4)
Modify the Bellman-Ford algorithm so that it sets v.d to $-\infty$ for all vertices v for which there is a negative-weight cycle on some path from the source to v.

解答:
flag = true;(算法最开始)
将第二个 for 循环进行 |V| 次

```
for i = 0 to |V|-1
    for each edge(u,v) ∈ G.E        if v.d > u.d + w(u,v)
            flag = false
            put v in set A;
            relax(u,v,w);
for each vertices in set A
    v.d = −∞
```

最后的返回值（在 for 循环结束后，即原算法的第 8 行）return flag

将 for 循环进行 |V| 次，路径有权重为负值的环路的点的 d 都会减小

题目 4 (TC 24.2-2)

Suppose we change line 3 of DAG-SHORTEST-PATH S to read

3 for the first |V|-1 vertices, taken in topologically sorted order

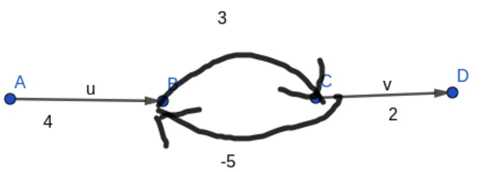
Show that the procedure would remain correct.

解答：

由于拓扑排序，排序后的一定没有从最后一个节点指向前面节点的边，所以在循环中去掉后无影响。

题目 5 (TC 24.3-2)

Give a simple example of a directed graph with negative-weight edges for which Dijkstra’s algorithm produces incorrect answers. Why doesn’t the proof of Theorem 24.6 go through when negative-weight edges are allowed?



不正确原因：在 24.2 式中，使用到了边的权值非负的条件 ($\sigma(s, y) \leqslant \sigma(s, u)$)，所以有负权值的边时结论不正确。

题目 6 (TC 24.3-4)

Professor Gaedel has written a program that he claims implements Dijkstra’s algorithm. The program produces v.d and $v.\pi$ for each vertex $v \in V$ .Give an O(V+E) time algorithm to check the output of the professor’s program. It should determine whether the d and π attributes match those of some shortest-paths tree.You may assume that all edge weights are nonnegative.

解答：

首先，检查这些节点是否构成一个树，遍历所有节点，根据 $v.\pi$ 将节点连接，并检差 v.d = u.d + w(u,v) 是否成立，时间开销为 O(|V|)

找到根节点 s，从 s 开始 BFS

在 BFS 过程中，首先需要增加一个数组记录是否可以从 s 到达每一个节点，时间开销为 O(|V|+|E|)

然后，对于每个定点，检查 $u.d + w(u, v) = min_{(z,v)\in E} z.d + w(z, v)$, 需要检查所有的

边，时间开销为 O(|E|)
总时间开销为 O(|V|+|E|)

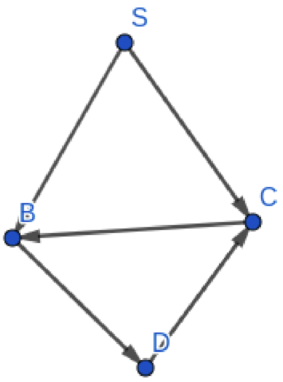
题目 7 (TC 24.3-7)
Let G=(V,E) be a weighted, directed graph with positive weight function $w \longrightarrow \{1, 2, ..., W\}$ for some positive integer W, and assume that no two vertices have the same shortest-path weights from source vertex s. Now suppose that we define an unweighted, directed graph $G^{'} = (V \bigcup V^{'}, E^{'})$ by replacing each edge $(u, v) \in E$ with w(u, v) unit-weight edges in series. How many vertices does $G^{'}$ have? Now suppose that we run a breadth-first search on $G^{'}$. Show that the order in which the breadth-first search of $G^{'}$ colors vertices in V black is the same as the order in which Dijkstra's algorithm the vertices of V from the priority queue when it runs on G.

解答：
节点的个数：
$|V| + \sum\limits_{foreachedgeinE} \omega(u, v) - |E|$=|V|-|E|+$\dfrac{W(1+W)}{2}$
u, v BFS $u.d$ $v.d$
$u.d \neq$ v.d，所以不妨 $u.d < v.d$，则可以得知，在得到的无权重的图中，u 距离 s（根节点）距离更近，所以 u 会先被遍历到，在 BFS 中，先被遍历到的点会先出队被染成黑色，所以 u 一定在 v 之前被染成黑色，由于 uv 是任意的所以 BFS 的染色顺序和 Dijkstra 算法的顺序是一样的。

题目 8 (TC 24.5-2)
Find a feasible solution or determine that no feasible solution exists for the follow-ing system of difference constraints:
$x_1 - x_2 \leqslant 4$
$x_1 - x_5 \leqslant 5$
$x_2 - x_4 \leqslant -6$
$x_3 - x_2 \leqslant 1$
$x_4 - x_1 \leqslant 3$
$x_4 - x_3 \leqslant 5$
$x_4 - x_5 \leqslant 10$
$x_5 - x_3 \leqslant -4$
$x_5 - x_4 \leqslant -8$

解答：
4,2,3,5,1,4 形成了负权值回路

题目 9 (TC 24.5-5)
Let G=(V,E) be a weighted, directed graph with no negative-weight edges. Lets $s \in V$ be the source vertex, and suppose that we allow $v.\pi$ to be the predecessor of v on any shortest path to v from source s if $v \in V - \{s\}$ is reachable from s,and NIL otherwise. Give an example of such a graph G and an assignment of π values that produces a cycle in G_π. (By Lemma 24.16, such an assignment cannot be produced by a sequence of relaxation steps.)



取每一条边的权重为 0，$B.\pi = C, C.\pi = D, D.\pi = B$，这样形成了一个环。

题目 10 (TC Problem 24-3)
Arbitrage is the use of discrepancies in currency exchange rates to transform one unit of a currency into more than one unit of the same currency. For example,suppose that1U.S. dollar buys49Indian rupees,1Indian rupee buys 2 Japanese yen, and1Japanese yen buys 0.0107 U.S.dollars. Then, by converting currencies,a trader can start with 1 U.S. dollar and buy 49x2x0.0107=1.0486 U.S.dollars,thus turning a profit of 4.86 percent.Suppose that we are given n currencies $c_1, c_2, ..., c_n$ and an $n \times n$ table R of exchange rates, such that one unit of currency c_i buys R[i,j] units of currency c_j
a.Give an efficient algorithm to determine whether or not there exists a sequence of currencies $< c_{i_1}; c_{i_2}, ..., c_{i_k} >$such that $R[i_1, i_2]R[i_2, i_3]...R[i_{k-1}, i_k]R[i_k, i_1] > 1$.
Analyze the running time of your algorithm.
b.Give an efficient algorithm to print out such a sequence if one exists. Analyze the running time of your algorithm.

解答：
a. 对所有的汇率表的值取负对数，将每一种货币看做一个顶点，将取对数后的汇率 $-lgR[i_k, i_j]$ 看做从 k 到 j 的一条有向边，权重为 $-lgR[i_k, i_j]$，然后在得到的图 G=(V,E) 上运行 Bellman-ford 算法，返回值为 true 则说明存在题目要求的货币序列。算法时间为 $O(n^3)$.
b.
先运行 a 中的 Bellman-ford 算法，然后在运行如下

```
for i = 0 to |V|
    for each edge (u,v) in E
    relax(u,v,w); ans=NIL; for each vertex u
    end = u;start = u;
    for i = 0 to |V|-1
        start = start.π
        if start == end ans = end; break（这里是跳出所有循环）；
start = ans;
do output ans
ans = ans.π
while(ans ≠ start)
```

后面算法先对所有边的 relax |V| 次，这样会使得在负权值环内的所有边的 π 都指向其在环内的顶点，然后对每个顶点枚举找到一个在环内的顶点，然后跳出循环，输出该顶点所在的负权值的环。

2 作业（选做部分）

题目 1 (TC Problem 24-2)

解答:

3 Open Topics

Open Topics 1 (Delta stepping algorithm)
参考资料:

- https://en.wikipedia.org/wiki/Parallel_single-source_shortest_path_algorithm
- Meyer, U.; Sanders, P. (2003-10-01). "Δ-stepping: a parallelizable shortest path algorithm".

Open Topics 2 (Radius stepping algorithm)
参考资料:

- https://en.wikipedia.org/wiki/Parallel_single-source_shortest_path_algorithm
- Blelloch, Guy E.; Gu, Yan; Sun, Yihan; Tangwongsan, Kanat (2016). "Parallel Shortest Paths Using Radius Stepping". Proceedings of the 28th ACM Symposium on Parallelism in Algorithms and Architectures - SPAA '16. New York, New York, USA: ACM Press: 443–454.

4 反馈