

离散数学（2023）作业 24 - 带权图与最短路

离散数学教学组

Problem 1

以下是 *Dijkstra* 算法的一个实现：

Algorithm 1 Dijkstra 算法

```
1: procedure DIJKSTRA( $G$ : 所有权都为正数的带权连通简单图)  $\{G$  有顶点  $a = v_1, v_2, \dots, z = v_n$ , 相应边上的  
   权值为  $w(v_i, v_j)\}$   
2:   for  $i := 1$  to  $n$  do  
3:      $L(v_i) := \infty$   
4:    $L(a) := 0$   
5:    $S := \emptyset$   
6:   while  $z \notin S$  do  
7:      $u :=$  不属于  $S$  的  $L(u)$  最小的一个顶点  
8:      $S := S \cup \{u\}$   
9:     for 所有不属于  $S$  的且属于  $N(u)$  的顶点  $v$  do  
10:       $L(v) := \min\{L(v), L(u) + w(u, v)\}$   
      {向  $S$  中添加带最小标记的顶点, 并且更新不在  $S$  中的顶点的标记}  
11: return  $L(z)$   $\{L(z) =$  从  $a$  到  $z$  的最短通路的长度 $\}$ 
```

证明: *Dijkstra* 算法第 8 行每次更新 S 时, 当前的 L 对于要加入的 u 都满足 $L(u)$ 等于 a 到 u 最短路的权值。

Problem 2

下面是 *Floyd* 算法的一个实现, 它求出了任意点对间所有边权的和最小的通路的长度, 请尝试修改该算法中相应的行, 解决下问题 (无需证明修改的正确性):

Algorithm 2 Floyd 算法

```
1: procedure FLOYD( $G$ : 带权简单图)  $\{G$  有顶点  $v_1, v_2, \dots, v_n$ , 相应边上的权值为  $w(v_i, v_j)\}$   
2:   for  $i := 1$  to  $n$  do  
3:     for  $j := 1$  to  $n$  do  
4:       if  $(v_i, v_j) \in E(G)$  then  
5:          $d(v_i, v_j) := w(v_i, v_j)$   
6:       else  
7:          $d(v_i, v_j) := \infty$   
8:   for  $i := 1$  to  $n$  do  
9:     for  $j := 1$  to  $n$  do  
10:      for  $k := 1$  to  $n$  do  
11:        if  $d(v_j, v_i) + d(v_i, v_k) < d(v_j, v_k)$  then  
12:           $d(v_j, v_k) := d(v_j, v_i) + d(v_i, v_k)$   
       $\{d(v_i, v_j)$  是在  $v_i$  与  $v_j$  之间的最短通路的长度, 为  $\infty$  时表示通路不存在 $\}$ 
```

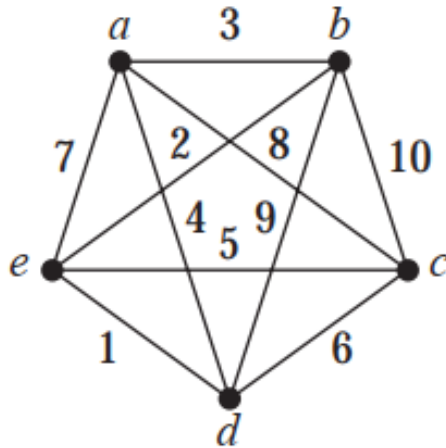
1. 对于任意边权大于 1 的图 G , 对于任意点对 a, z , 试求 a 到 z 的通路所有边的权值乘积最小可以是多少; (下同)
2. 定义一条通路的“强度”为通路上权最小的边的权值 (例如, 由权值依次为 3, 5, -2, 1 的边组成的通路强度为 -2), 对于任意点对 a, z , 试求 a 到 z 的最强的通路的强度可以是多少。

Problem 3

若边的权可以为负数, *Dijkstra* 算法能否正确求出最短路? 若可以, 请给出证明; 若不能, 请举出一个反例并分析说明。

Problem 4

基于所有哈密顿回路的权值选择出总权值最小的回路来解决下图的旅行商问题。(注: $ad=4$, $bd=9$, $be=2$, $ac=8$, $ec=5$ 。需要给出总权值最小的回路和其权值, 若答案不唯一, 需全部写出)



Problem 5

描述如何调用已知的求两点间最短通路长度的算法解决下列求简单加权 (权均为正数) 连通图上的有限制的最短通路长度问题:

1. 求从顶点 v_i 出发到达 v_j , 且经过顶点 v_k 的最短通路长度 (为保证通路最短, 可以经过同一个顶点多次);
2. 求从顶点 v_i 出发到达 v_j , 且不经顶点 v_k 的最短通路长度;
3. 求从顶点 v_i 出发, 先经过顶点 v_k , 再到达顶点 v_j 的最短通路长度 (即在到达 v_k 前不得经过顶点 v_j);