

Problem Set 26: 带权图与最短路

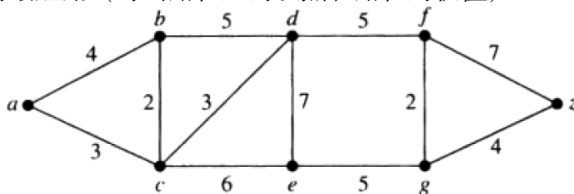
提交截止时间：6 月 3 日 10:00

注：本 Problem Set 中涉及算法证明相关的问题（Problem 2, Problem 3）的目的在于启发思考，不作为考试的内容；请大胆尝试，做错也不影响本次作业的成绩。

Problem 1

求以下带权图的下列顶点对之间的最短路（写出路径上的顶点和路径的权值）。

- a) a 和 d
- b) a 和 f
- c) c 和 f
- d) b 和 z



Problem 2

Algorithm 1 弗洛伊德算法

procedure FLOYD(G : 带权简单图)

$\{G$ 有顶点 v_1, v_2, \dots, v_n 和权 $w(v_i, v_j)$, 其中若 (v_i, v_j) 不是边, 则 $w(v_i, v_j) = \infty\}$

for $i := 1$ **to** n **do**

for $j := 1$ **to** n **do**

$d(v_i, v_j) := w(v_i, v_j)$

for $i := 1$ **to** n **do**

for $j := 1$ **to** n **do**

for $k := 1$ **to** n **do**

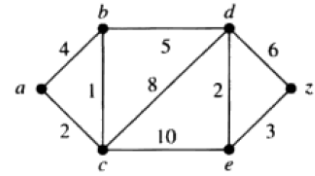
if $d(v_j, v_i) + d(v_i, v_k) < d(v_j, v_k)$ **then**

$d(v_j, v_k) := d(v_j, v_i) + d(v_i, v_k)$

$\{d(v_i, v_j)$ 是在 v_i 与 v_j 之间的最短通路的长度 $\}$

a) 用弗洛伊德算法求右图中带权图里所有顶点对之间的距离.

b) 证明: 当 *Floyd* 最外层的循环执行完 $i = p$ 时, $d(v_j, v_k)$ 表示从 v_j 到 v_k 且路径上中间顶点都在集合 $\{v_1, v_2, \dots, v_p\}$ 中的路径的最短距离.



Problem 3

下面是 *Dijkstra* 算法的一个实现, 它求出了从 a 出发到 z 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, 解决下列问题:

Algorithm 2 Dijkstra 算法

procedure DIJKSTRA(G : 所有权都为正数的带权连通简单图)

$\{G$ 有顶点 $a = v_1, v_2, \dots, z = v_n$ 和权 $w(v_i, v_j)$, 其中若 (v_i, v_j) 不是 G 的边, 则 $w(v_i, v_j) = \infty\}$

for $doi := 1$ **to** n

$L(v_i) := \infty$

$L(a) := 0$

$S := \emptyset$

while $doz \notin S$

$u :=$ 不属于 S 的 $L(u)$ 最小的一个顶点

$S := S \cup \{u\}$

for 所有不属于 S 的顶点 v **do**

$L(v) := \min\{L(v), L(u) + w(u, v)\}$

$\{\text{向 } S \text{ 中添加带最小标记的顶点, 并且更新不在 } S \text{ 中的顶点的标记}\}$

return $L(z)$ $\{L(z) =$ 从 a 到 z 的最短通路的长度 $\}$

a) 对于任意边权大于 1 的图 G , 对于给出的点 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? (无需证明, 下同)

b) 对于任意边权非负的图 G , 对于给出的点 a, z , 试求 a 到 z 的任意通路最短的边最长可以有多长.

Problem 4

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 5

证明或反驳: 对于权值为正的简单连通图 G , 在已知图上任意两点间最短路长度的前提下, 可以构建出 G .

Problem 6

利用已有最短路算法，解决下列求简单加权（权均为正数）连通图上的最短通路长度的问题：

- a) 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）；
- b) 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；
- c) 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 ）。

Problem 7

求下图中以 A 为源点到图中其他所有点的最短路。请按顺序写出路径上的点，并给出各路径的权值。

