

HW7

PB19111713钟颖康

1.

$O(V^2)$ 。因为在邻接矩阵中需要对每个顶点检验是否与当前顶点相邻。

2.

①使用 $tarjan$ 算法得到图中所有的强连通分量

②每个强连通分量均用一个顶点替换，遍历所有边更新 E 得到新的图 G'

③对 G' 进行拓扑排序判定，若每轮删除入度为0的点及其相关边后有且仅有一个新的入度为0的点，则是单连通图，否则不是

3.

1~|V|:

①初始化 $O(1)$

②计数排序 $O(E)$

③ $FIND - SET$ 、 $UNION$ 、 $MAKE - SET$ 操作共计 $O((V + E)\alpha(V))$

④不相交集合操作的时间代价 $O(E\alpha(V))$

总运行时间 $O(E \lg E)$

1~W:

①初始化 $O(1)$

②计数排序 $O(E)$

③ $FIND - SET$ 、 $UNION$ 、 $MAKE - SET$ 操作共计 $O((V + E)\alpha(V))$

④不相交集合操作的时间代价 $O(E\alpha(V))$

总运行时间 $O(E \lg E)$

4.

1~|V|:

根据边的权值将边放进至多 $|V|$ 个常数列表中，常数列表中增删只需常数时间复杂度 $O(1)$

①设置 key 所需时间 $O(V)$

②每次 $EXTRACT - MIN$ 所需时间 $O(1)$ ，总计 $O(V)$

③每次 *DECREASE - KEY* 所需时间 $O(1)$, 总计 $O(E)$

总运行时间为 $O(V) + O(V) + O(E) = O(V + E)$

1~W:

优先队列使用 *van Emde Boas* 树结构

①设置 *key* 所需时间 $O(V)$

②每次 *EXTRACT - MIN* 所需时间 $O(\lg(\lg W))$, 总计 $O(V \lg(\lg W))$

③每次 *DECREASE - KEY* 所需时间 $O(\lg(\lg W))$, 总计 $O(E \lg(\lg W))$

总运行时间为 $O(V) + O(V \lg(\lg W)) + O(E \lg(\lg W)) = O((V + E) \lg(\lg W))$