计算机学院《算法设计与分析》
(2020 年秋季学期)

第四次作业

作业提交截止时间：2020 年 12 月 20 日 23 : 55

1 对下面的每个描述，请判断其是正确或错误，或无法判断正误。对于你判为错误的描述，请说明它为什么是错的。(每小题 5 分，共 20 分)

1. NP-$hard \subseteq NP$；
2. 对某问题 $X \in NP$ 而言，若可以证明规约式 3-$SAT \leq_p X$，则 $X \in NPC$；
3. $P \neq NP$；
4. 所有 NP 完全问题均无法在多项式时间内被解决。

2 最小点集问题 (20 分)

给定一个包含 n 个点的连通有向图 $G = (V, E)$，节点编号为 $1, 2, \cdots, n$，请设计算法找出最小的点集 $U \subseteq V$，使得对所有点 $v \in V$，均存在某点 $u \in U$，使得图中存在一条从 u 到 v 的路径。如果这样的点集有多个，求出任意一个即可。此外，请分析该算法的时间复杂度。

例如，给定一个包含 $n = 6$ 个点的图，边集 $E = \{(1, 2), (1, 3), (3, 6), (4, 5), (5, 3)\}$。可以发现，在该图中从 1 出发可到达 $2, 3, 6$，从 4 出发可到达 $3, 5, 6$。因此，选择点集 $U = \{1, 4\}$ 即可满足条件。

3 项目排序问题 (20 分)

公司有 n 个项目和 m 个小组，项目和小组都从 1 开始编号。每个项目可能有两种情况：一是没有归属，二是由某个小组负责。我们用 $group[i]$ 代表第 i 个项目所属的小组，如果这个项目无需任何小组负责，那么 $group[i] = 0$，。

现需要对这些项目制定完成顺序，并返回排序后的项目列表，要求如下：

1. 同一小组的项目，排序后在列表中彼此相邻；
2. 项目之间存在一定的依赖关系，用 $pre[i]$ 表示。其含义为在进行第 i 个项目前，应该完成的项目集合。

如果存在多个解决方案，只需要返回其中任意一个即可；如果没有合适的解决方案，就返回一个空列表。请设计算法解决该问题并分析其时间复杂度。

例如，给定 $n = 8$ 个项目，$m = 2$ 个小组，项目归属 $group = [0, 0, 2, 1, 1, 2, 1, 0]$，其含义为第 3，第 6 个项目由小组 2 负责；第 4，第 5，第 7 个项目由小组 1 负责；其他项目无需任何小组负责。项目依赖关系如下：$pre[2] = \{7\}, pre[3] = \{6\}, pre[4] = \{7\}, pre[5] = \{4, 7\}$。第 1 和第 6 个项目不依赖于任何其他项目，因此这里略去。

在此情况下，一个可行的项目完成顺序为 $[7, 4, 5, 2, 6, 3, 1, 8]$，其中项目 $4, 5, 7$ 同属小组 1，项目 $3, 6$ 同属小组 2，在列表中相邻；项目 7 在 2 前，6 在 3 前，7 在 4 前，$4, 7$ 在 5 前，满足项目依赖关系。

4　方程式求解问题 (20 分)

给定 n 个变量 $a_i (1 \le i \le n)$，和 m 个方程式，其中第 k 个方程式以三元组 $(i_k, j_k, v_k)(1 \le i_k, j_k \le n)$ 的形式给出，其含义为 $a_{i_k}/a_{j_k} = v_k$。请设计算法，根据已知的 m 个方程式求解目标式子 $a_x/a_y (1 \le x, y \le n)$ 的值，并分析该算法的时间复杂度。

可以假设输入总是有效的，且除法运算中不会出现除数为 0 的情况。

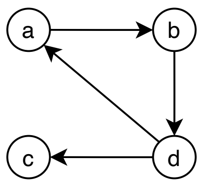
例如：对给定的 3 个变量 a_1, a_2, a_3，已知 $m = 2$ 个方程式 $a_1/a_2 = 3, a_2/a_3 = 2$（对应的三元组分别为 $(1, 2, 3)$ 和 $(2, 3, 2)$），现要求取目标式子 a_1/a_3 的值 $(x = 1, y = 3)$。则根据上述两个方程式可推出，$a_1/a_3 = 6$。

5　传递闭包问题 (20 分)

给定一个包含 n 个节点的有向图 $G = (V, E)$，其传递闭包定义为一个 n 阶布尔矩阵 $T = \{t_{ij}\}$，其中矩阵第 i 行 $(i \le i \le n)$ 第 j 列 $(1 \le j \le n)$ 的元素 t_{ij} 表示图中是否存在从 i 到 j 的路径。如果从第 i 个顶点到第 j 个顶点之间存在一条有向路径，则 t_{ij} 为 1；否则 t_{ij} 为 0。

现给定一个有向图的邻接矩阵 A，请设计算法求出其传递闭包 T 并分析该算法的时间复杂度。

例如，对于有向图



其邻接矩阵为 $A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 \end{pmatrix}$，求出的传递闭包为 $T = \begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{pmatrix}$。