

# 《数据结构与算法》第十五次作业练习题

开始时间 12/22/2023 6:10:00 PM

结束时间 12/26/2023 11:59:00 PM

答题时长 6109分钟

答卷类型 标准答案

总分 49

## 判断题

得分：暂无 总分：8

1-1 拓扑序一定是唯一的。(2分)

☐ T ☒ F

1-2 拓扑排序的有向图中，最多存在一个环路。(2分)

☐ T ☒ F

1-3 若有向图G存在拓扑排序序列，则G一定是弱连通的。(2分)

☐ T ☒ F

1-4 P 是顶点 S 到 T 的最短路径，如果该图中的所有路径的权值都加 1，P 仍然是 S 到 T 的最短路径。(2分)

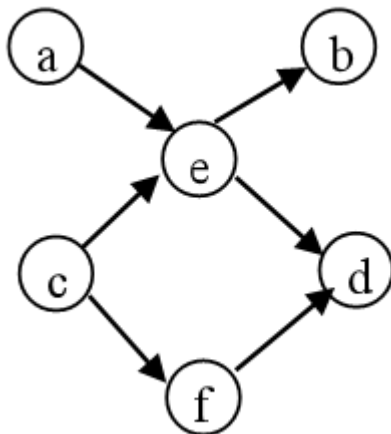
☐ T ☒ F

## 单选题

得分：暂无 总分：6

2-1 有向图G如下，则其拓扑序列不可能的是 ( )

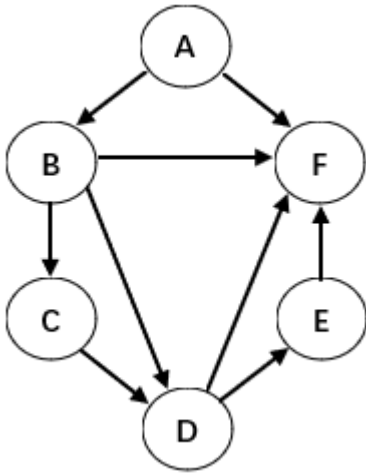
(2分)



- ☐ A. acefbd  
☐ B. cfaebd  
☒ C. cefabd  
☐ D. acfedb

2-2 给定如下有向图，该图的拓扑有序序列的个数是：

(2分)



- ☒ A. 1
- ☐ B. 2
- ☐ C. 3
- ☐ D. 4

2-3 已知有向图G=(V, E), 其中  $V = \{v1, v2, v3, v4, v5, v6\}$ ,  $E = \{<v1,v2>, <v1,v4>, <v2,v6>, <v3,v1>, <v3,v4>, <v4,v5>, <v5,v2>, <v5,v6>\}$ 。G的拓扑序列是: (2分)

- ☒ A. v3, v1, v4, v5, v2, v6
- ☐ B. v3, v4, v1, v5, v2, v6
- ☐ C. v1, v3, v4, v5, v2, v6
- ☐ D. v1, v4, v3, v5, v2, v6

编程题

得分: 暂无    总分: 35

7-1 任务拓扑排序 (10分)

一个工程被分解成n个子任务，编号为0至n-1。要完成整个工程需要完成所有的子任务。其中一些子任务必须先于另外一些子任务被完成。给定各子任务之间的先后关系，请编写程序给出一个合理的任务完成顺序，若工程不可行，程序亦能识别。

输入格式:

输入第一行为两个整数n和e，均不超过100。n表示子任务数。接下来e行，表示已知的两个子任务间的先后关系，每行为两个整数a和b，表示任务a必须先于任务b完成。

输出格式:

若工程不可行（一些子任务以自己为先决条件），输出“unworkable project”；若工程可行，输出为1行整数，每个整数后一个空格，为n个子任务的编号，表示子任务的完成顺序，如果有多种可能的顺序，则输出字典序最小者。  
注：字典序，即对象在字典中的顺序。对于两个数字序列，从第一个数字开始比较，当某一个位置的数字不同时，该位置数字较小的序列，字典序较小，例如1 2 3 9比1 2 4 5小，1 2 8 9比1 2 10 3小。

输入样例1:

```
3 2
0 1
1 2
```

输出样例1:

```
0 1 2
```

输入样例2:

```
3 3
0 1
1 2
2 0
```

输出样例2:

```
unworkable project
```

7-2 任务调度的合理性（25分）

假定一个工程项目由一组子任务构成，子任务之间有的可以并行执行，有的必须在完成了其它一些子任务后才能执行。“任务调度”包括一组子任务、以及每个子任务可以执行所依赖的子任务集。

比如完成一个专业的所有课程学习和毕业设计可以看成一个本科生要完成的一项工程，各门课程可以看成是子任务。有些课程可以同时开设，比如英语和C程序设计，它们没有必须先修哪门的约束；有些课程则不可以同时开设，因为它们有先后的依赖关系，比如C程序设计和数据结构两门课，必须先学习前者。

但是需要注意的是，对一组子任务，并不是任意的任务调度都是一个可行的方案。比如方案中存在“子任务A依赖于子任务B，子任务B依赖于子任务C，子任务C又依赖于子任务A”，那么这三个任务哪个都不能先执行，这就是一个不可行的方案。你现在的任务是写程序判定任何一个给定的任务调度是否可行。

输入格式:

输入说明：输入第一行给出子任务数 $N$ （ $\leq 100$ ），子任务按 $1\sim N$ 编号。随后 $N$ 行，每行给出一个子任务的依赖集合：首先给出依赖集合中的子任务数 $K$ ，随后给出 $K$ 个子任务编号，整数之间都用空格分隔。

输出格式:

如果方案可行，则输出1，否则输出0。

输入样例1:

```
12
0
0
2 1 2
0
1 4
1 5
2 3 6
1 3
2 7 8
1 7
1 10
1 7
```

输出样例1:

```
1
```

输入样例2:

```
5
1 4
2 1 4
2 2 5
1 3
0
```

输出样例2:

```
0
```