



# 数据结构与算法实验

## 实验17：拓扑排序 – 第三次上机作业

彭振辉

中山大学人工智能学院

2023年秋季学期

# 实验目的和要求



- 巩固所学的有关图的基本知识;
- 熟练掌握对**有向无环图的拓扑排序算法**;
- 掌握如何**应用拓扑排序算法**解决各种实际问题。
- 拓扑排序的实现以库文件方式实现，不得在测试主程序中直接实现;
- 程序有较好可读性，各运算和变量的命名直观易懂，符合关键工程要求;
- 程序有适当的注释。

## 正确性：

- **20分 = 5 (task1) + 5 (task2) + 10 (task3)**
- 助教会检查你在每个任务中输入测试用例得到的结果截图，如全部用例都得到正确的输出，该task得满分，否则每错一个用例按相应任务说明扣分。
- 助教会检查**所有**同学代码，检查是否有抄袭或未实现的情况。如有，相应的task抄袭者和被抄袭者皆不得分

## 时效性：

- **当堂课结束前提交不扣分**
- **当天结束前提交扣2分**
- 第二天结束前提交扣4分，第三天结束前提交扣6分，第四天结束前提交扣8分，第五天及以后提交扣10分。



# 提交格式和方式

## 格式：

- 请把“第十七周作业”文件夹（需存放你**写入代码的cpp文件、编译生成的exe文件**，和包含“整体实现参考 + 2-3句简要体会（如教训、思路、拓展应用等） + 所有测试用例结果的截图”的pdf/doc文档）压缩成一个文件，命名为：**学号\_姓名\_第十七周作业.zip**



210235\_彭振辉\_  
第三周作业.zip

## 提交方式（具体操作教程在最后几页）：

- 对分易提交（当堂课或者课后提交均可用该方式）

# 截图样例和其在文档中的标记



## 实验任务1/2:

输入1:

5 8 13 22 50 100 -1

3 11 77 101 220 -1

输入2:

88 -1

3 7 86 88 110 -1

截图方式:

Windows: Shift +  
Command + S

或者用“截图工具”  
这个软件

输入3: ...

输入4: ...

在文档中标记好: 任务1\_测试1

在文档中标记好: 任务2\_测试2.jpg



# 实验任务1

- 根据输入构建有向无环图，输出所有的拓扑序列
  - 以邻接链表为存储结构
  - 输入点、边信息，对输入信息进行检测，判断是否为无环图
  - 设计递归回溯的拓扑排序算法，输出所有拓扑序列

示例:

Input:

6  
1 2 3 4 5 6

2  
2  
3  
2  
4  
5  
2  
4  
5  
1  
6  
1  
6  
0

请输入所有顶点数:6  
请依次输入每个顶点的名称:  
1 2 3 4 5 6

示例输入说明

请输入第1个顶点的指向顶点的数量:2  
请输入第1个邻接点的编号:2  
请输入第2个邻接点的编号:3  
请输入第2个顶点的指向顶点的数量:2  
请输入第1个邻接点的编号:4  
请输入第2个邻接点的编号:5  
请输入第3个顶点的指向顶点的数量:2  
请输入第1个邻接点的编号:4  
请输入第2个邻接点的编号:5  
请输入第4个顶点的指向顶点的数量:1  
请输入第1个邻接点的编号:6  
请输入第5个顶点的指向顶点的数量:1  
请输入第1个邻接点的编号:6  
请输入第6个顶点的指向顶点的数量:0

Output:

经拓扑排序检测，输入信息符合无环图  
该图所有拓扑序列为：

1 2 3 4 5 6  
1 2 3 5 4 6  
1 3 2 4 5 6  
1 3 2 5 4 6

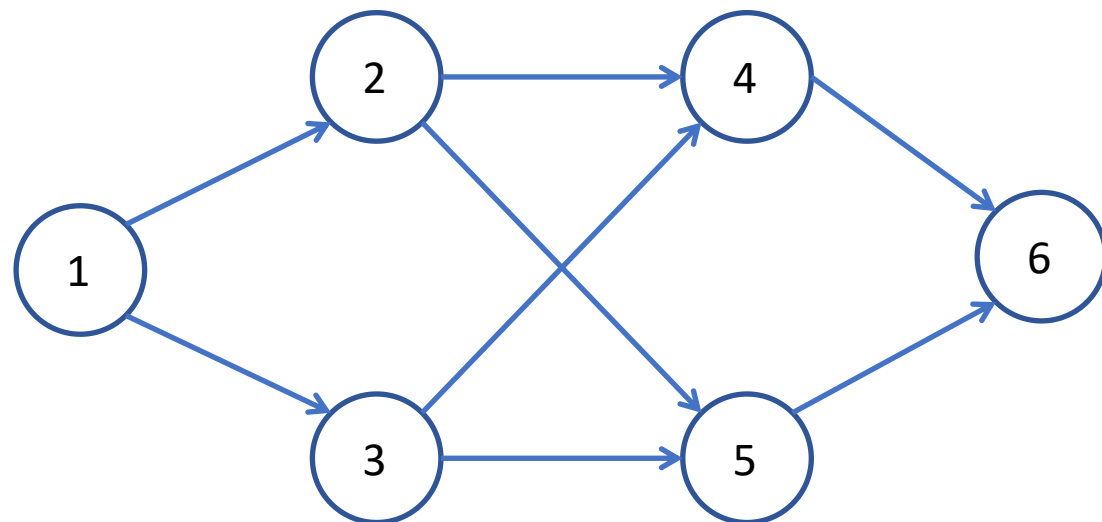
# 实验任务1



示例:

Input:

6  
1 2 3 4 5 6  
2  
2  
3  
2  
4  
5  
2  
4  
5  
1  
6  
1  
6  
0



Output:

经拓扑排序检测，输入信息符合无环图  
该图所有拓扑序列为：

1 2 3 4 5 6  
1 2 3 5 4 6  
1 3 2 4 5 6  
1 3 2 5 4 6



# 实验任务1 测试用例

输入1:

6  
1 2 3 4 5 6  
3  
2  
3  
4  
1  
3  
2  
5  
6  
1  
5  
1  
6  
0

输入2:

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

(同学们自己验证其输出)

输入3:

7  
1 2 3 4 5 6 7  
6  
3  
2  
4  
5  
7  
6  
5  
3  
4  
6  
7  
7  
4  
7  
6  
5  
4  
3  
5  
6  
7  
2  
7  
6  
1  
7  
0

说明：为了实验任务1可以得到**满分5分**，你需要提交**三个截图**，每张截图分别包含以上**输入用例和对应的输出**，我们会检查你每个用例是否得到正确的输出，**每错一个，扣2分，扣完为止**。如并未对任务1做出具体实现，该任务1得分为0。

正确输出:

经拓扑排序检测，输入信息符合无环图  
该图所有拓扑序列为：

1 2 3 4 5 6  
1 2 4 3 5 6  
1 4 2 3 5 6





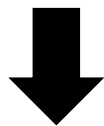
## 实验任务2

### 示例:

9 12  
0 1 6  
0 2 4  
0 3 5  
1 4 1  
2 4 1  
3 5 2  
5 4 0  
4 6 9  
4 7 7  
5 7 4  
6 8 2  
7 8 4

一个项目由若干个任务组成，任务之间有先后依赖顺序。项目经理需要设置一系列里程碑，在每个里程碑节点处检查任务的完成情况，并启动后续的任务。现给定一个项目中各个任务之间的关系，请你计算出这个项目的**最早完工时间**

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



Impossible

– 输入：第一行给出两个正整数：项目里程碑的数量  $N(\leq 100)$  和任务总数  $M$ 。这里的里程碑从 0 到  $N-1$  编号。随后  $M$  行，每行给出一项任务的描述，格式为“任务起始里程碑 任务结束里程碑 工作时长”，三个数字均为非负整数，以空格分隔

– 输出：如果整个项目的安排是合理可行的，在一行中输出最早完工时间；否则输出"Impossible"

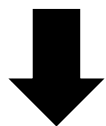
↓  
18

## 实验任务2

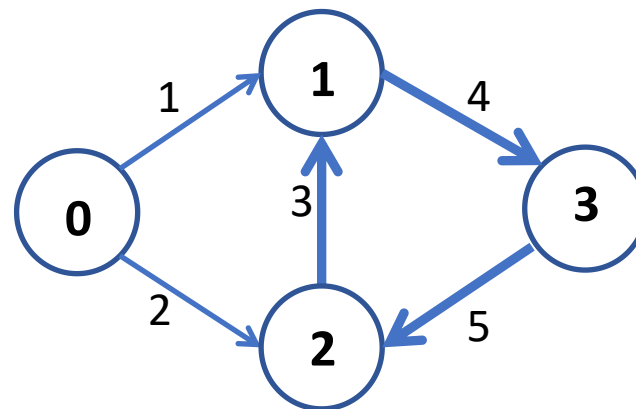
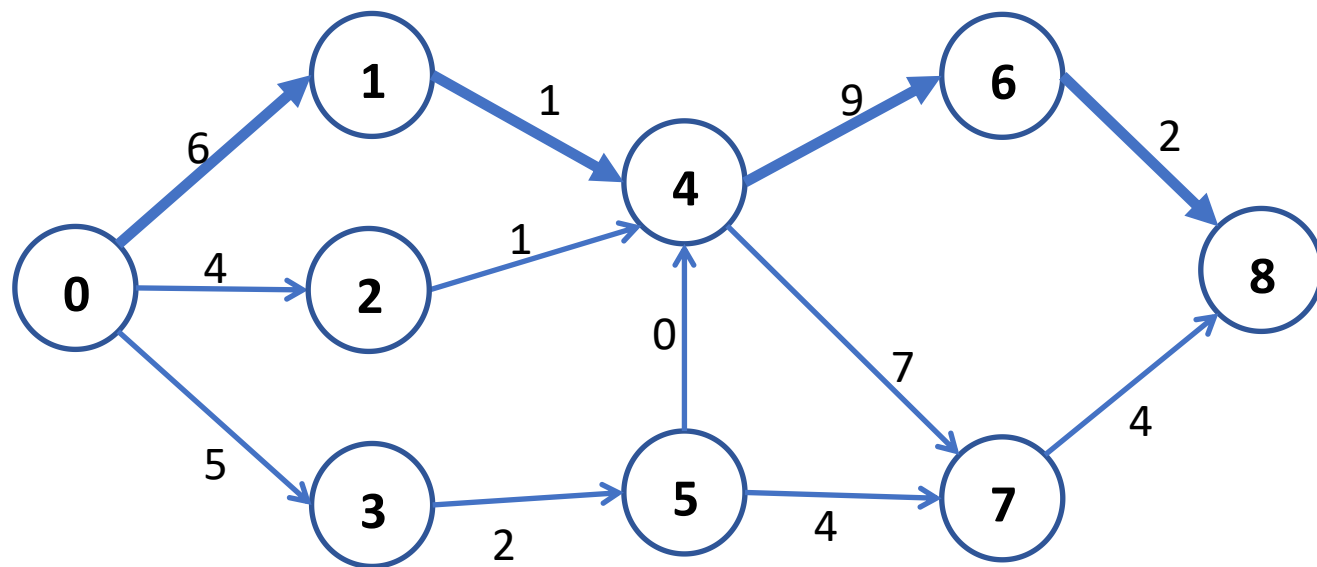


示例:

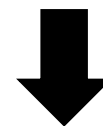
9 12  
0 1 6  
0 2 4  
0 3 5  
1 4 1  
2 4 1  
3 5 2  
5 4 0  
4 6 9  
4 7 7  
5 7 4  
6 8 2  
7 8 4



18



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



Impossible



## 实验任务2 测试用例

输入1:

8 10  
0 1 5  
0 3 7  
0 5 9  
1 3 8  
1 7 15  
2 6 9  
2 5 5  
3 6 9  
4 7 7  
5 6 1

输入2:

7 9  
0 1 1  
0 2 2  
0 6 6  
1 3 2  
1 5 8  
2 6 10  
3 4 1  
3 5 8  
4 6 11

输入3:

6 6  
0 1 2  
0 4 3  
0 5 15  
1 5 20  
2 4 3  
3 5 11

正确输出: 22

(同学们自己验证其输出)

说明: 为了实验任务2可以得到**满分5分**, 你需要提交**三个截图**, 每张截图分别包含以上**输入用例和对应的输出**, 我们会检查你每个用例是否得到正确的输出, **每错一个, 扣2分, 扣完为止**。如并未对任务2做出具体实现, 该任务2得分为0。



## 实验任务3（简而言之：关键路径问题）

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（环），这就是一个不可行的方案。

任务调度问题中，若还给出了完成每个子任务需要的时间，则可以算出完成整个工程需要的最短时间。在这些子任务中，有些任务可以推迟几天完成，也不会影响全局的工期；但有些任务必须准时完成，否则整个项目的工期就要因此延误，这种任务就叫“关键活动”。

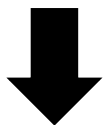
**请编写程序判定一个给定的工程项目的任务调度是否可行；如果该调度方案可行，则计算完成整个工程项目需要的最短时间，并输出所有的关键活动**

## 实验任务3



示例:

7 8  
1 2 4  
1 3 3  
2 4 5  
3 4 3  
4 5 1  
4 6 6  
5 7 5  
6 7 2



17  
1->2  
2->4  
4->6  
6->7

– 输入：第1行给出两个正整数 $N(\leq 100)$ 和 $M$ ，其中 $N$ 是任务交接点(衔接相互依赖的两个子任务的节点，例如：若任务2要在任务1完成后才开始，则两任务之间必有一个交接点)的数量。**交接点按1~ $N$ 编号， $M$ 是子任务的数量，依次编号为1~ $M$ 。**随后 $M$ 行，每行给出了3个正整数，分别是**该任务开始和完成涉及的交接点编号以及该任务所需的时间**，整数间用空格分隔

– 输出：**若任务调度不可行，则输出0；否则第1行输出完成整个工程项目需要的时间，第2行开始输出所有关键活动，每个关键活动占一行，按格式“ $V \rightarrow W$ ”输出，其中 $V$ 和 $W$ 为该任务开始和完成涉及的交接点编号。关键活动输出的顺序规则是：任务开始的交接点编号小者优先，起点编号相同时，与输入时任务的顺序相反**

# 实验任务3



示例:

7 8

1 2 4

1 3 3

2 4 5

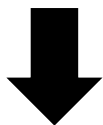
3 4 3

4 5 1

4 6 6

5 7 5

6 7 2



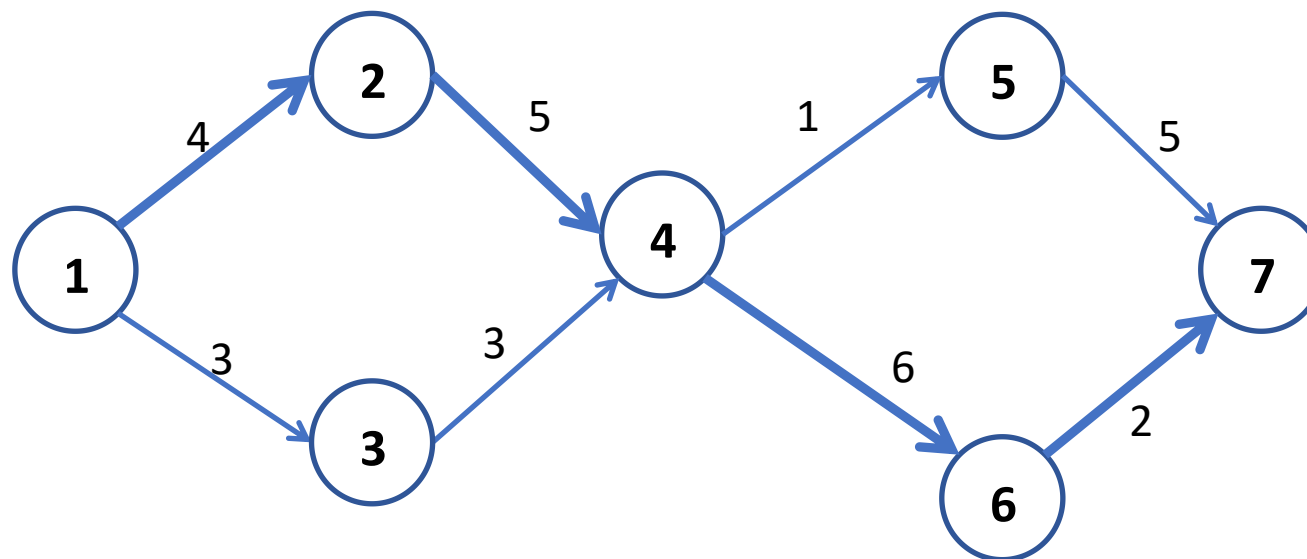
17

1->2

2->4

4->6

6->7





## 实验任务3 测试用例

输入1:

6 3  
1 2 3  
1 3 5  
2 6 8

正确输出:

11  
1->2  
2->6

输入2:

8 9  
1 2 5  
1 3 7  
1 5 10  
2 3 1  
2 4 7  
2 8 15  
3 7 9  
3 8 8  
4 5 1

输入3:

8 10  
1 2 5  
1 4 7  
1 6 9  
2 4 8  
2 8 15  
3 7 9  
3 6 5  
4 7 9  
5 8 7  
6 7 1

(同学们自己验证其输出)

说明: 为了实验任务3可以得到**满分10分**, 你需要提交**三个截图**, 每张截图分别包含以上**输入用例和对应的输出**, 我们会检查你每个用例是否得到正确的输出, **每错一个, 扣4分, 扣完为止**。如并未对任务3做出具体实现, 该任务3得分为0。





# 对分易提交方法

<https://www.duifene.com/Home.aspx>

微信扫码登录对分易

对分易

让教师更轻松 让学生更主动

登录

免费注册

用户登录

登录名

密码

☐ 记住密码 [忘记登录密码](#)

登录

手机微信扫一扫登录

对分易 教学平台

首页 上门培训 关于我们 新闻动态 活动专区 注册 登录





# 对分易提交方法

对分易

教学平台

欢迎您, 彭振辉 同学

[切换为老师](#)

备考神器

帮助

个人中心

点进数据结构与算法的班级, 点击作业

班级学生

课程资源

作业

微信消息

考勤

讨论区

分组

成绩册

在线练习

课堂提问

调查问卷

投票

活动

教学评价

弹幕讨论

学生互测

互动直播

考试

# 对分易提交方法



首页 > 课程：数据结构与算法 > 作业列表

21级人工智能

点进相应的作业编号，去完成，把压缩文件上传成附件，提交

截止时间：2022-09-07 17:04

已交53人

去完成

单次作业

写作业

作业文本：

限制8000个字

0/8000

作业文件：

①上传附件

单个文件大小限制为100MB，文件名称请不要超过100个字，视频文件请转成H.264编码的MP4格式。

☐ 视频保存后自动转为教师可以在线播放的格式（H264编码的MP4格式）

提交作业