

1240301、1240307 《程序设计实践 II》任务书

1 总体要求

2 任务

2.1 题目 1：电脑迷宫鼠

2.1.1 基础要求

(1) 概述：用 java 面向对象程序设计语言，设计和实现一电脑鼠走迷宫的软件程序，即一个假想的小车能在图示的迷宫中根据设定的起始点和终点自主寻找路径。本综合实践分成两部分：第一部分为算法设计和实现部分，第二部分为界面展现部分。每一部分又分为三个不同层次的要求，学生可以根据自己的能力来完成相应的部分。

(2) 第一部分：算法设计和实现部分

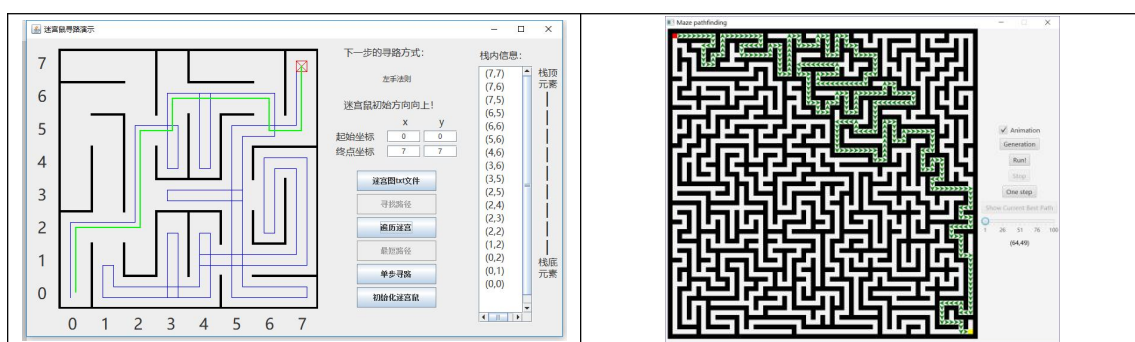
- 迷宫地图生成算法的设计和实现
 - 自动生成迷宫：根据迷宫生成算法自动生成一定复杂度的迷宫地图。
 - 手动生成迷宫：根据文件中存储的固定数据生成迷宫地图。
- 单路径寻找算法的设计与实现：找出迷宫中一条单一的通路。
- 迷宫遍历算法的设计与实现：遍历迷宫中所有的可行路径。
- 最短路径计算算法的设计与实现：根据遍历结果，找出迷宫中所有通路中的最短通路。

(3) 第二部分：界面展示部分

- 生成迷宫地图界面的设计与实现：根据生成的迷宫地图，用可视化的界面展现出来。
- 界面布局的设计与实现：根据迷宫程序的总体需求，设计和实现合理的界面布局。
- 相关迷宫生成过程和寻路算法在界面上的展现：将迷宫程序中的相关功能，跟界面合理结合，并采用一定的方法展现给用户，如通过动画展示等。

(4) 总体任务要求

- 具有判断通路和障碍的功能；
- 走不通具备返回的能力（路径记忆）；
- 能够寻找最短路径；
- 程序不仅要实现相关算法，还需要具备基本的界面操作功能，如下图所示。



(5) 任务分解

- 迷宫的生成：手动生成或自动生成
- 寻路：从任意给定点走到另外给定点
- 遍历：遍历整个迷宫
- 寻优：计算最短路径（计算等高表，按路径行规定走）
- 相关界面设计和编程

(6) 实现过程要充分利用面向对象的思想对程序整体架构进行设计，对相关的功能进行封装。

2.1.2 相关数据结构

(1) 常用数据结构概述

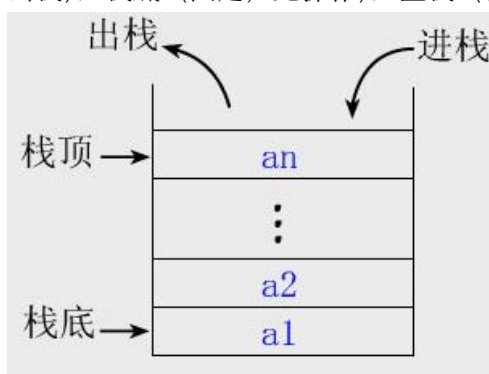
- 寻路：栈（存储走过的路径）、二维数组
- 遍历：栈、二维数组（记录整个迷宫的墙信息，每个数组元素为一个结构体结点包含一个迷宫坐标点上右下左的墙信息）
- 寻优：队列（建立等高表）、栈（记录最优路径）

(2) 数据结构—栈（如下图所示）

操作特点：先进后出(Last In First Out, LIFO)

结构特点：只允许一端插入、查看和删除，不允许对另一端的操作

概念：栈顶（入栈、出栈）；栈底（固定，无操作）；空栈（栈中没有元素）



(3) Java 中的栈实现

Java 中类 `Stack<E>` 实现了栈的相关功能，继承自 `Vector<E>`，构造方法：`Stack()`，表示创建一个空堆栈，常用方法如下图所示。

boolean	<u>empty</u> ()	测试堆栈是否为空。
<u>E</u>	<u>peek</u> ()	查看堆栈顶部的对象，但不从堆栈中移除它。
<u>E</u>	<u>pop</u> ()	移除堆栈顶部的对象，并作为此函数的值返回该对象。
<u>E</u>	<u>push</u> (<u>E</u> item)	把项压入堆栈顶部。
int	<u>search</u> (Object o)	返回对象在堆栈中的位置，以 1 为基数。

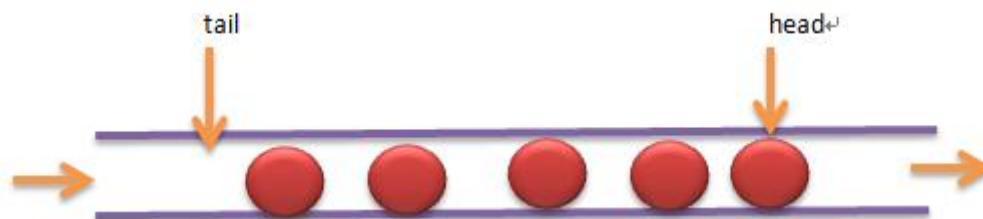
(4) 数据结构—队列 (如下图所示)

操作特性: 先进先出(First In First Out, FIFO)。

结构特点: 只允许一端入队, 另一端查看和删除。

概念: 队头 (出队); 队尾 (入队); 空队列 (队列中没有元素)。

主要类型: 顺序队列 (队列、循环队列)、链式队列、两端队列



(5) Java 中队列的实现

Java 中的队列常用操作通过接口 **Queue** 来实现, 它继承自 **Collection** 接口。**LinkedList** 实现了 **Queue** 接口, 实现了对队列的相关操作, 常用操作如下图所示。

add	增加一个元素	如果队列已满, 则抛出一个 IllegalArgumentException 异常
remove	移除并返回队列头部的元素	如果队列为空, 则抛出一个 NoSuchElementException 异常
element	返回队列头部的元素	如果队列为空, 则抛出一个 NoSuchElementException 异常
offer	添加一个元素并返回 true	如果队列已满, 则返回 false
poll	移除并返回队列头部的元素	如果队列为空, 则返回 null
peek	返回队列头部的元素	如果队列为空, 则返回 null
put	添加一个元素	如果队列满, 则阻塞
take	移除并返回队列头部的元素	如果队列为空, 则阻塞

2.1.3 相关算法

(1) 算法—回溯

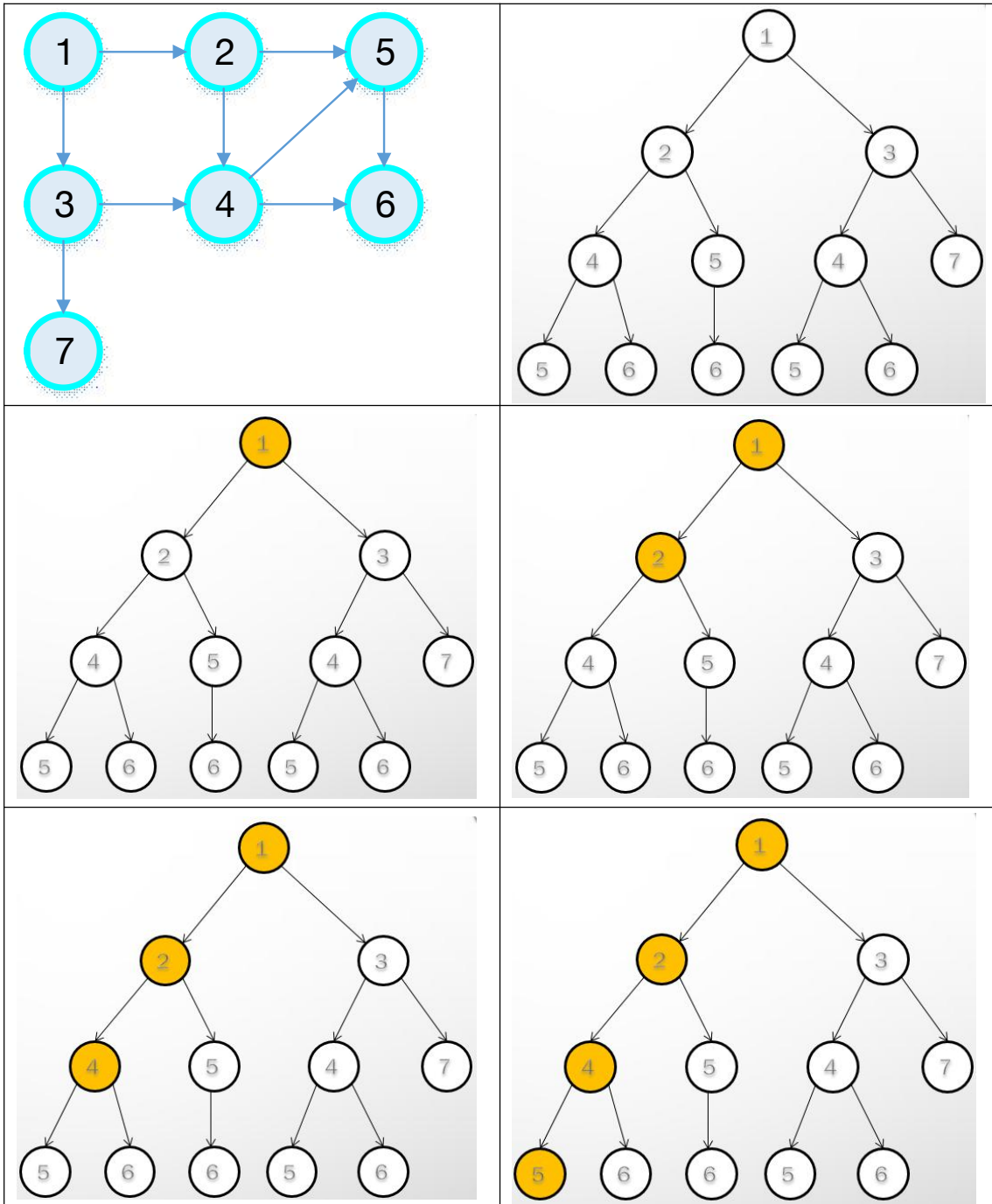
基本思想和步骤: 回溯算法的基本思想就是深度优先遍历, 基本步骤如下:

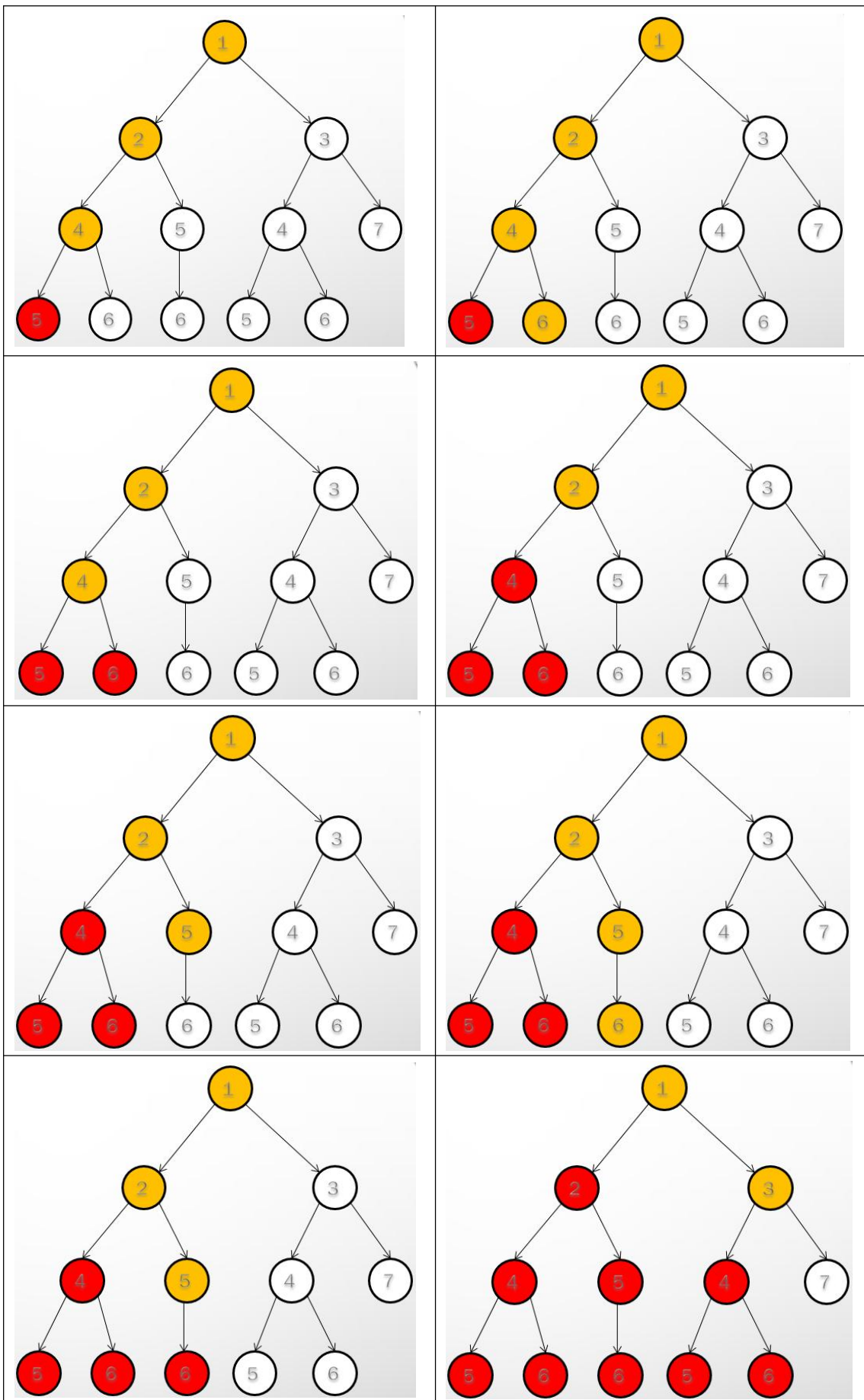
步骤一: 先将根结点作为活动结点;

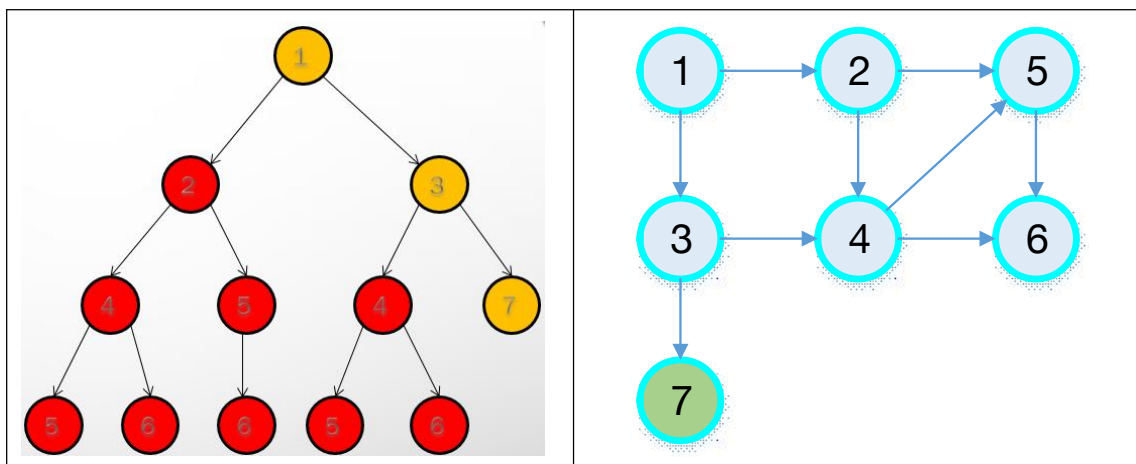
步骤二: 向活动结点的子结点扩展, 若其存在, 子结点作为新的活结点, 若不存在可扩展子结点, 将当前活动结点设为不可扩展结点, 其父节点作为当前的活动结点;

步骤三: 重复步骤二, 直至找到所求结点, 输出路径, 或者到没有可扩展结点时结束,

输出 null。如下图所示。







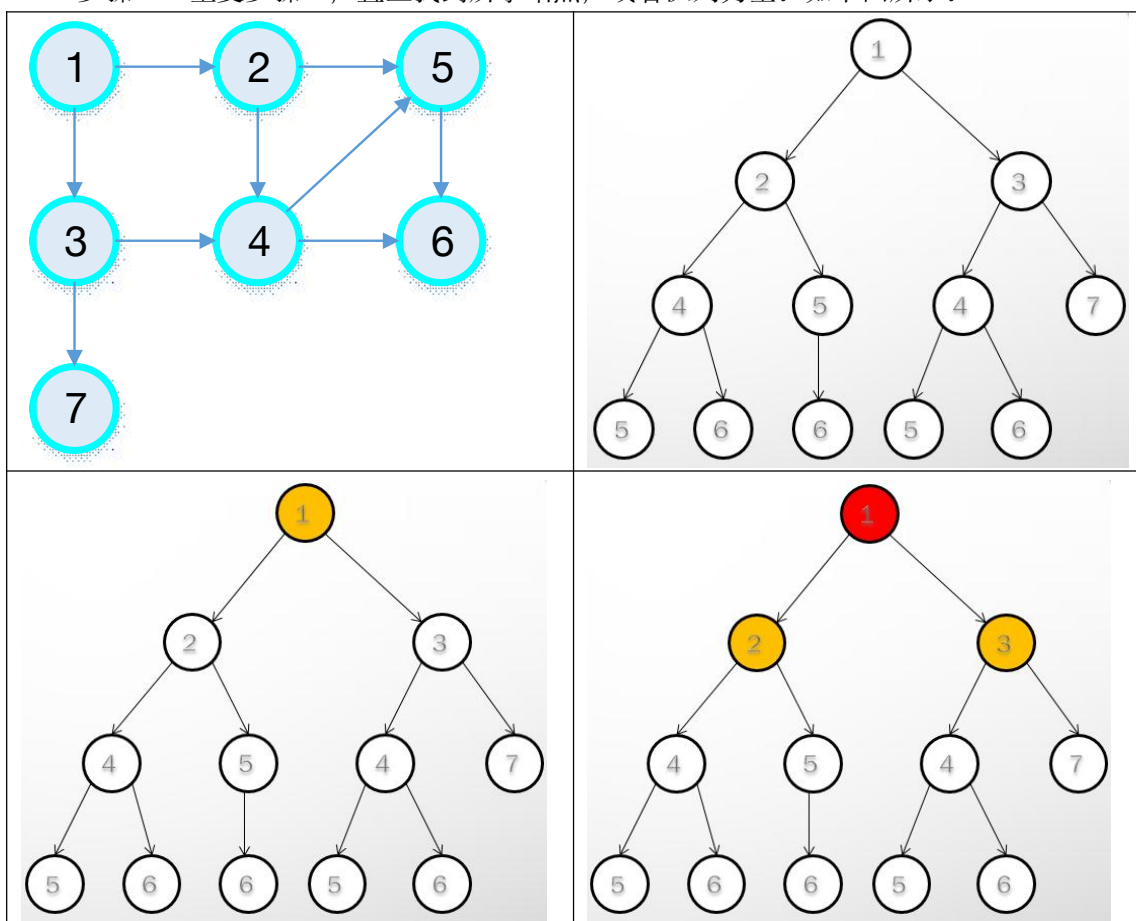
(2) 算法-广度优先遍历

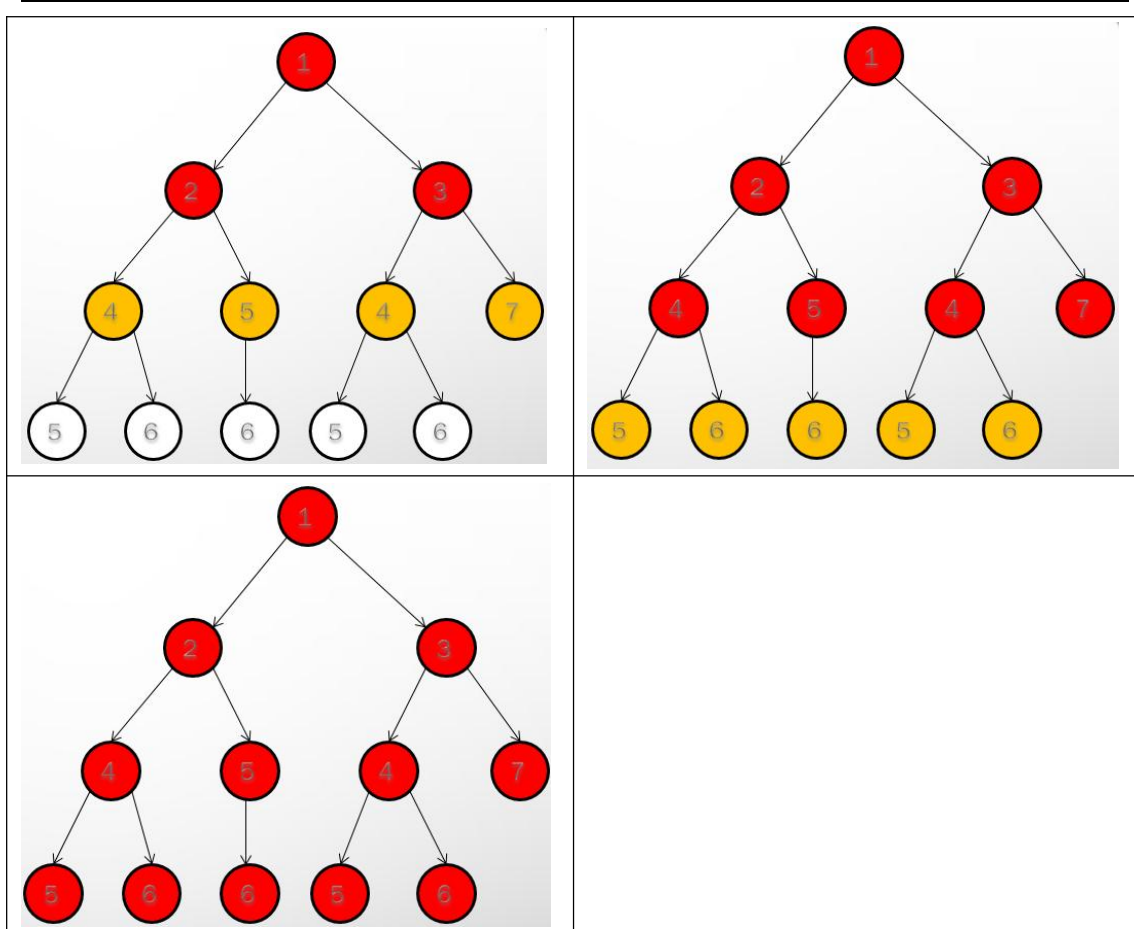
基本思想和步骤：横向优先遍历，基本步骤如下：

步骤一：先将跟结点入队列；

步骤二：取队头元素，遍历队头元素的所有子结点，并将所有子结点加入队列；

步骤三：重复步骤二，直至找到所求结点，或者队列为空。如下图所示。





(3) 寻路过程中可以遵行的几种法则

- 右手法则：小车在搜过程中有两个以上的搜索方向时，优先选择向右转，其次是向前行进，最后才考虑向左转弯。
- 左手法则：小车在搜过程中有两个以上的搜索方向时，优先选择向左转，其次是向前行进，最后才考虑向右转弯。
- 向心法则：把整个迷宫地图分解为左下，右下，左上，右上四个均等的区域，在不同的区域中选择不同的转弯策略，使得小车始终向着迷宫的中心靠近，这样就可以以最快的速度接近终点，总体上说，遍历迷宫时此方法更科学有效。

(4) 自动生成迷宫地图参考资料

- ① <https://blog.csdn.net/u011714944/article/details/37994111>
- ② <https://blog.csdn.net/yutianzuijin/article/details/52078340> (递归分割法)
- ③ https://blog.csdn.net/weixin_31787547/article/details/113411564 (prim 算法)
- ④

https://www.bilibili.com/video/BV1xp4y1r7Ng?p=1&share_medium=iphone&share_plat=ios&share_source=COPY&share_tag=s_i×tamp=1613300655&unique_k=UCDHPH
(prim 算法视频讲解)

2.2 题目 2：管理信息系统类

2.2.1 基础要求

- (1) 采用 java 基础知识 (JDBC) 实现关系型数据库数据 (建议采用 MySQL 数据库) 的增加、删除、修改和查找等操作功能。
- (2) 要求采用 java 可视化编程包实现信息系统类的相关操作界面。
- (3) 可以选择的题目比如：学生选课管理信息系统，停车管理信息系统，餐厅收银管理信息系统等，但不限于以上题目。
- (4) 题目选择好后需要按照实际需求分析每个系统的需求，并形成需求文档，系统实现的功能应尽量跟实际需求接近。
- (5) 实现过程要充分利用面向对象的思想对程序整体架构进行设计，对相关的功能进行封装。

2.2.2 其它要求

- (1) 关系型数据库尽量采用 mysql，关于 mysql 的简单使用可以参照资料：https://www.bilibili.com/video/BV1Vt411z7wy?spm_id_from=333.337.search-card.all.click
- (2) java 操作 mysql 数据库采用 jdbc 技术，关于 jdbc 的简单实用可以参考资料：https://www.bilibili.com/video/BV1wy4y1h74w?spm_id_from=333.337.search-card.all.click

2.3 题目 3：java 网络编程基础点对点聊天系统

2.3.1 基础要求

- (1) 采用 java 网络编程技术实现一个点对点的局域网聊天和文件传输系统。
- (2) 要求采用 java 可视化编程包实现相关的操作界面。
- (3) 可以选择性的采用数据库或文件存储相关的信息。
- (4) 为了实现系统操作上的异步性，本题目需要采用 java 多线程编程技术。
- (5) 本作品运行的时候包括一个服务器和多个客户端，客户端和客户端之间可以聊天，发送文件等。
- (6) 可以根据个人能力选择性实现是否支持离线消息，文件断点续传等功能。
- (7) 实现过程要充分利用面向对象的思想对程序整体架构进行设计，对相关的功能进行封装。

2.3.2 其它要求

- (1) 关系型数据库尽量采用 mysql，关于 mysql 的简单使用可以参照资料：

https://www.bilibili.com/video/BV1Vt411z7wy?spm_id_from=333.337.search-card.all.click

(2) java 操作 mysql 数据库采用 jdbc 技术, 关于 jdbc 的简单实用可以参考资料:

https://www.bilibili.com/video/BV1wy4y1h74w?spm_id_from=333.337.search-card.all.click

(3) java 网络编程基础参考资料:

https://www.bilibili.com/video/BV1j54y1b7qv?spm_id_from=333.337.search-card.all.click

2.4 题目 4: 简单游戏开发

2.4.1 基础要求

(1) 请从以下游戏列表中选择一款小游戏完成

- 贪吃蛇游戏:
- 俄罗斯方块游戏
- 捕鱼达人游戏
- 扫雷游戏
- 斗地主游戏
- 坦克大战游戏

(2) 要求采用 java 可视化编程包实现相关的操作界面。

(3) 要求游戏具有游戏过程中可以中断并存盘退出, 下一次继续游戏的功能。

(4) 实现过程要充分利用面向对象的思想对程序整体架构进行设计, 对相关的功能进行封装。

2.4.2 其它要求

(1) 贪吃蛇游戏参考资料:

https://www.bilibili.com/video/BV1gX4y1579N?spm_id_from=333.337.search-card.all.click

(2) 俄罗斯方块游戏参考资料:

https://www.bilibili.com/video/BV1xM4y157qX?spm_id_from=333.337.search-card.all.click

(3) 捕鱼达人游戏参考资料:

https://www.bilibili.com/video/BV1e5411s7KR?spm_id_from=333.337.search-card.all.click

(4) 扫雷游戏参考资料:

https://www.bilibili.com/video/BV1gY4y1k7vA?spm_id_from=333.337.search-card.all.click

(5) 斗地主游戏参考资料:

https://www.bilibili.com/video/BV15q4y1M7Pi?spm_id_from=333.337.search-card.all.click

(6) 坦克大战游戏参考资料:

https://www.bilibili.com/video/BV1zP4y1b7gw?spm_id_from=333.337.search-card.all.click

3 选题规则

- (1) 如果选择题目 1, 则不需要选择其他的题目, 只完成一道题即可;
- (2) 如果不选择题目 1, 则题目 4 为必选题, 题目 2 和 3 中二选一, 共需要完成两道题。

4 时间/地点安排

另见通知。

5 提交资料要求

(1) 提交资料包括**打印的纸质文档**和**电子文档**, 电子文档以班级为单位提交 U 盘, 每个学生的资料放在一个文件夹内, 文件夹名称为学生的学号, 里面包括该生纸质文档的电子版本和综合实践的程序; 纸质文档学习委员收齐后统一交给老师, 并同时提交 U 盘。

(2) 资料截止提交时间另行通知。

(3) 文档内容要求:

- **电子文档和程序严禁抄袭, 如发现雷同, 同时取消两个人的综合实践成绩**
- 电子文档根据自己的实际情况如实撰写, 自己没有实现的功能和算法不能写进文档。
- 程序中尽量多写注释, 要求注释的行数要超过程序行数的 1/3。
- 纸质文档双面打印后用订书机装订提交即可。
- 文档中必须包括以下内容:
 - 题目功能的描述和分析
 - 功能模块的分解描述
 - 关键问题的解决方案描述
 - 主要类的 UML 类图 (包括类图和继承关系图)
 - 文档模板详见附件 1
 - 文档中不能粘贴源代码
- 本次综合实践任务每个人独立完成, 不分组

6 考核

(1) 申请优秀的同学, 需要单独提出申请, 由指导老师单独检查, 并录制一个不超过 15 分钟的视频讲解。视频讲解中应包含:

- 系统总体思想介绍
- 系统算法介绍
- 系统功能演示（配合讲解）
- 系统亮点强调

(2) 不申请优秀同学，由教师课堂上随机组合 5 个人为一个答辩组。每个答辩组中同学根据每个同学答辩情况，按照对应的评分点评分后（评分表参见文件《程序设计实践 II 原始成绩计算表_学生评分表.xlsx》）进行组内排名。

(3) 根据学生的评分和排名情况，老师随机抽取每个答辩组中的学生参加教师组织的答辩，以被选中答辩学生的最终得分为基准，整个答辩组的所有同学的成绩在此基础上根据第二步所得到的排名上下浮动。比如，老师选中的是该组中排名第一的同学，如果该同学的在老师那里的答辩成绩为中等，那么该答辩组所有成绩不超过中等。

(4) 综合实践的总成绩=系统（检查时候的系统演示和代码质量）*0.6+沟通表达（检查时候的表达能力和问题回答能力）*0.1+课程报告成绩*0.3。

附件 1 课程报告模板

另外参见文档《程序设计实践 II 报告模板.docx》