

《数 据 结 构》

实习大纲



实习内容：

- 1、马踏棋盘游戏开发；（25 分）
- 2、迷宫问题求解；（25 分）
- 3、查找算法比较。（25 分）
- 4、内部排序算法比较；（25 分）

实习安排：

- 1、2022 年 6 月 27 日（10 学时，完成实习内容 1、内容 2 的程序；
- 2、2022 年 6 月 28 日（10 学时：完成实习内容 3 的程序；
- 3、2022 年 6 月 29 日（10 学时：完成实习内容 4 的程序；完成实习报告检查）

实习要求：

- 1、在完成实习内容题目的同时可以自选一些有意义并且和《数据结构》相关的综合性、设计性题目，经老师批准并定分值后作为实习题；
- 2、程序要求有详细的注释，没有注释的程序不予检查；
- 3、程序应主要由自己独立完成；
- 4、实习期间缺席一次扣十分，缺席三次则本门课计为 0 分；
- 5、实习完成后提交实验报告（包括每题的程序实现）。

实习检查：

- ① 每天点名两次；
- ② 程序完成后由教师检查、提问，并于实习结束前将程序压缩后提交到教师指定的位置。压缩文件名为：2020*****（???）——其中，*为学号后七位，??? 为学生姓名，压缩文件内要有明确的题号。如：2020*****_3 作为题目文件夹，还需要包括实习报告，如 2020*****_BG.pdf。

实习内容

1、马踏棋盘游戏开发；（25 分）

【问题描述】

设计一个国际象棋的马踏遍棋盘的演示程序。

【基本要求】

将马随机放在国际象棋的 8×8 棋盘 $\text{Board}[0 \sim 7][0 \sim 7]$ 的某个方格中，马按走棋规则进行移动。要求每个方格只进入一次，走遍棋盘上全部 64 个方格。编制非递归程序，求出马的行走路线，并按求出的行走路线，将数字 1, 2, ..., 64 依次填入一个 8×8 的方阵，输出之。

【测试数据】

由自己指定。可自选指定一个马的初始位置 (i, j) ， $0 \leq i, j \leq 7$ 。

【实现提示】

	0	1	2	3	4	5	6	7
0			8		1			
1		7				2		
2				H				
3		6				3		
4			5		4			
5								
6								
7								

上图显示了马位于方格 $(2, 3)$ 时，8 个可能的移动位置：

一般来说，当马位于位置 (i, j) 时，可以走到下列 8 个位置之一： $(i-2, j+1)$ ， $(i-1, j+2)$ ， $(i+1, j+2)$ ， $(i+2, j+1)$ ， $(i+2, j-1)$ ， $(i+1, j-2)$ ， $(i-1, j-2)$ ， $(i-2, j-1)$

但是，如果 (i, j) 靠近棋盘的边缘，上述有些位置可能超出棋盘范围，成为不允许的位置。8 个可能位置可以用两个一维数组 $\text{HTry1}[0 \sim 7]$ 和 $\text{HTry2}[0 \sim 7]$ 来表示：

	0	1	2	3	4	5	6	7
HTry1	-2	-1	1	2	2	1	-1	-2

	0	1	2	3	4	5	6	7
HTry2	1	2	2	1	-1	-2	-2	-1

位于 (i, j) 的马可以走到的新位置是在棋盘范围内的 $(i + \text{HTry1}[h], j + \text{HTry2}[h])$ ，其中 $h=0, 1, \dots, 7$ 。

每次在多个可走位置中选择其中一个进行试探，其余未曾试探过的可走位置必须用适当结构妥善管理，以备试探失败时的“回溯”（悔棋）使用。

【选做内容】

- （1）求出从某一起点出发的多条以至全部行走路线。
- （2）探讨每次选择位置的“最佳策略”，以减少回溯的次数。
- （3）演示寻找行走路线的回溯过程。

2、迷宫问题求解; (25 分)

【问题描述】

以一个 $M \times N$ 的长方阵表示迷宫, 0 和 1 分别表示迷宫中的通路和障碍。设计一个程序, 对任意设定的迷宫, 求出一条从入口到出口的通路, 或得出没有通路的结论。

【基本要求】

首先实现一个以链表作存储结构的栈类型, 然后编写一个求解迷宫的非递归程序。求得的通路以三元组 (i, j, d) 的形式输出, 其中: (i, j) 指示迷宫中的一个坐标, d 表示走到下一坐标的方向。如: 对于下列数据的迷宫, 输出一条通路为: $(1, 1, 1)$, $(1, 2, 2)$, $(2, 2, 2)$, $(3, 2, 3)$, $(3, 1, 2)$, ...。

【测试数据】

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

⇒

迷宫的测试数据如下: 左上角 $(1, 1)$ 为入口, 右下角 $(8, 9)$ 为出口。

【实现提示】

计算机解迷宫通常用的是“穷举求解”方法, 即从入口出发, 顺着某一个方向进行探索, 若能走通, 则继续前进; 否则沿着原路退回, 换一个方向继续探索, 直至出口位置, 求得一条通路。假如所有可能的通路都探索到而未能到达出口, 则所设定的迷宫没有通路。

可以二维数组存储迷宫数据, 通常设定入口的下标为 $(1, 1)$, 出口点的下标为 (n, n) 。为处理方便起见, 可在迷宫的四周加一圈障碍。对于迷宫中任一位置, 均可约定有东、南、西、北四个方向可通。

【选做内容】

- (1) 编写递归形式的算法, 求得迷宫中所有的可能的通路;
- (2) 以方阵形式输出迷宫及其通路。

3、查找算法比较。(25 分)

【问题描述】

查找是在大量的信息中寻找一个特定的信息元素，在计算机应用中，查找是常用的基本运算，在教课书中，介绍了常见的各种查找算法，给出了各种查找算法的时间复杂度分析或大概执行时间。试通过随机数据比较各算法的关键字比较次数，以取得直观感受。其实二分查找、插值查找以及斐波那契查找都可以归为一类——插值查找。插值查找和斐波那契查找是在二分查找的基础上的优化查找算法。各种查找算法的时间复杂度分析结果只给出了算法执行时间的阶，或大概执行时间。试通过随机数据比较各算法的关键字比较次数和关键字移动次数，以取得直观感受。

【基本要求】

- (1) 对以下 6 种常用的内部排序算法进行比较：顺序查找，折半查找，分块查找，插值查找，斐波那契查找，二叉树查找。
- (2) 待查找表的表长不小于 100；其中的数据要用伪随机数产生程序产生；至少要用 5 组不同的输入数据作比较；比较的指标为有关关键字参加的比较次数。
- (3) 最后要对结果作出简单分析，包括对各组数据得出结果波动大小的解释。

【测试数据】

由随机数产生器生成。

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
srand((int)time(NULL)); 设定随机数种子
```

```
rand()%100; 产生 0-99 的随机数。
```

【实现提示】

主要工作是设法在已知算法中的查找适当位置的关键字，并分析比较次数。程序还可以考虑几组数据的典型性，如：正序、逆序。注意采用分块调试的方法。

4、内部排序算法比较；（25 分）

【问题描述】

在教科书中，各种内部排序算法的时间复杂度分析结果只给出了算法执行时间的阶，或大概执行时间。试通过随机数据比较各算法的关键字比较次数和关键字移动次数，以取得直观感受。

【基本要求】

- （1）对以下 6 种常用的内部排序算法进行比较：冒泡排序，直接插入排序，简单选择排序，快速排序，希尔排序，堆排序。
- （2）待排序表的表长不小于 100；其中的数据要用伪随机数产生程序产生；至少要用 5 组不同的输入数据作比较；比较的指标为有关关键字参加的比较次数和关键字的移动次数(关键字交换计为 3 次移动)。
- （3）最后要对结果作出简单分析，包括对各组数据得出结果波动大小的解释。

【测试数据】

由随机数产生器生成。

```
#include <stdlib.h>
```

```
#include <time.h>
```

```
Srand((int)time(NULL)); 设定随机数种子
```

```
rand()%100; 产生 0-99 的随机数。
```

【实现提示】

主要工作是设法在已知算法中的适当位置插入对关键字的比较次数和移动次数的计数操作。程序还可以考虑几组数据的典型性，如：正序、逆序和不同程度的乱序。注意采用分块调试的方法。