杨静
jyang@cs. ecnu. edu. cn

理科大楼B703

评分比例

期末书面考试：60%
期中考试：20%
平时作业课堂提问等：5%
实践课评分：15%

- 上机练习完成情况
- 上机报告
- 上机考

关于上机实践

实践课机房时间比上一学年减少一半，
练习量没有减少。

关于教材

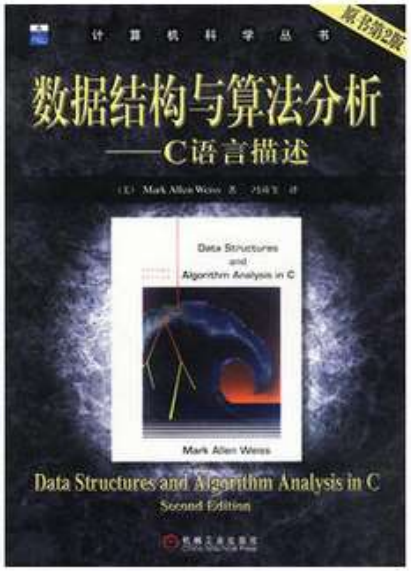
本课程主要内容来源

教材优点：
知识点覆盖全面（大夏学堂上有扫描版本）

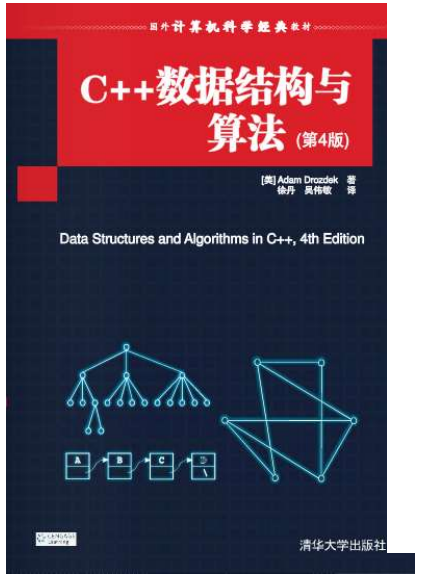
教材缺点：
描述语言版本陈旧
程序段无注释
缺少编程技巧及总体规划方面的讨论

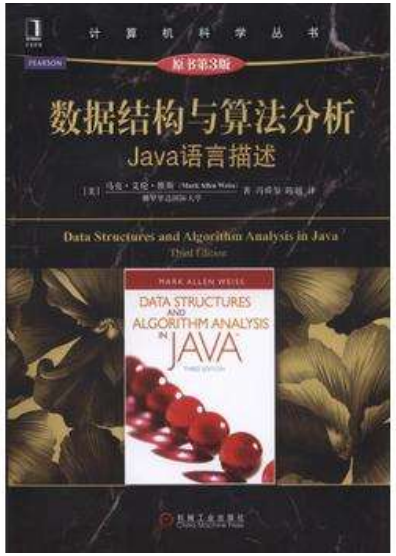
关于教材

各类参考教材：


计 算 机 科 学 丛 书
数据结构与算法分析
——C语言描述
Data Structures and Algorithm Analysis in C
Second Edition


清华大学 计算机系列教材
数据结构
（C语言版）
清华大学出版社


C++数据结构与算法（第4版）
Data Structures and Algorithms in C++, 4th Edition
清华大学出版社


计 算 机 科 学 丛 书
原书第3版
数据结构与算法分析
Java语言描述
Data Structures and Algorithm Analysis in Java
DATA STRUCTURES AND ALGORITHM ANALYSIS IN JAVA

课程内容

1～4章：线性关系的数据结构：
线性表，栈，队列，串，
排序算法

5～7章：非线性关系的数据结构：树，二叉树，图

第 0 章
绪 论

主要知识点

课程主要知识点分布

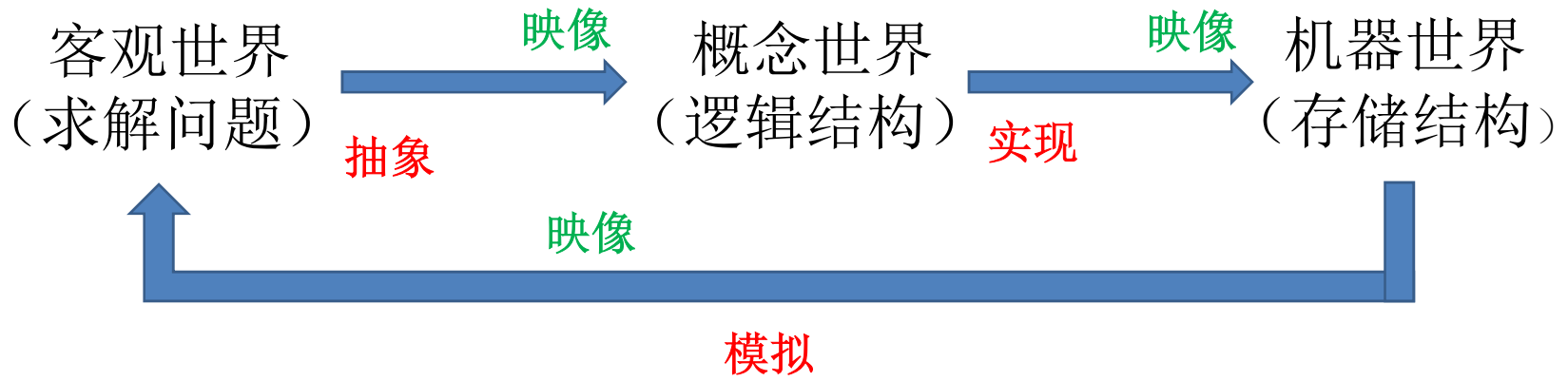
	存储	基础算法与应用
线性表（队列，栈）	顺序 链式 其他（压缩，索引，散列）	插入，删除，查找 （查找算法：顺序，二分，分块，Hash） 排序：插入，选择，冒泡，希尔，合并，快速，基数
串	顺序	查找
树	链式：非线性存储 顺序：线性表示	遍历
二叉树	顺序，链式	遍历 穿线树，穿线排序 查找树，平衡树，B-树，Tire结构，红黑树 堆，堆排序 应用：解答树，背包问题，皇后问题
图	顺序：邻接矩阵 链式：邻接表	图的遍历，求图的联通分量 生成树，最小代价生成树 最短路径，拓扑排序，关键路径

第 0 章
绪 论

基本概念和术语

1 引言

- 求解问题



1 引言

- 处理过程

- 核心
  - 数据结构
  - 算法
  - 程序 = 数据结构 + 算法

问题一

- 对任意输入的8个整数，按由小到大的顺序显示出来

第 0 章
绪 论

问题一

- 对任意输入的8个整数，按由小到大的顺序显示出来

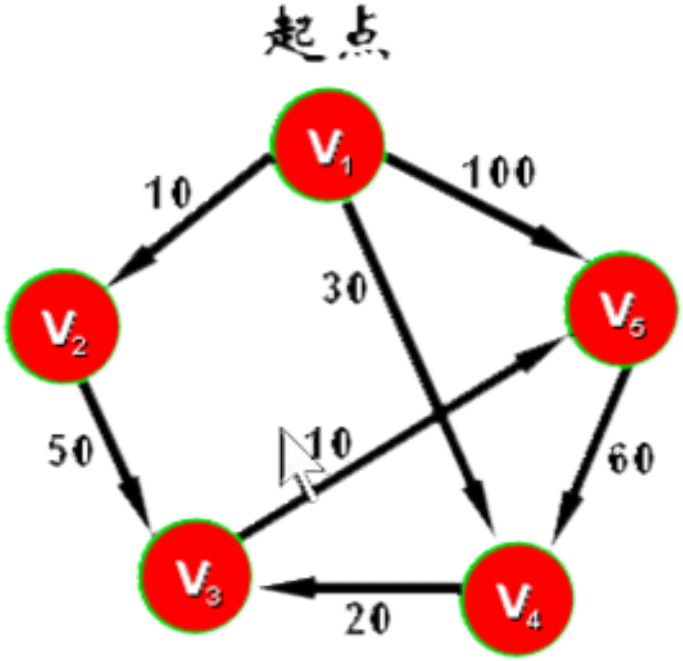
	1	2	3	4	5	6	7	8
	49	38	65	97	76	13	27	49
第一趟	38	49	65	76	13	27	49	97
第二趟	38	49	65	13	27	49	76	97
第三趟	38	49	13	27	49	65	76	97
.....								
第七趟	13	27	38	49	49	65	76	97

- 表示：序列　49、38、65、97、76、13、27、49
- 存储：数组　int a[8];
- 处理：排序　相邻元素比较、交换（冒泡排序法）

问题二

- 现有5个城市的连通图，结点表示城市，有向边表示城市间公路，边上的标数表示公路的公里数。请问由城市V1 出发分别到达各个城市的最短路径。

- 从V1 → V5 的各种情况



问题一、二涉及的问题

- 如何描述数据、以及数据与数据之间的关系?

-- 表

一维：数字序列

二维：新生信息

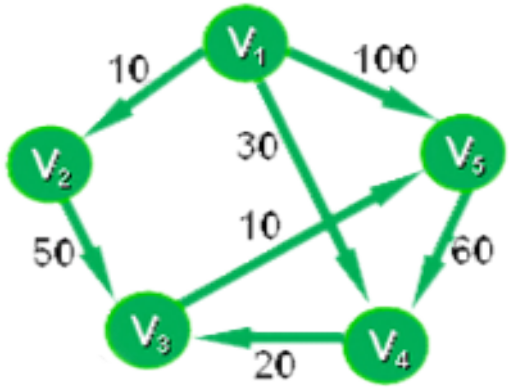
学号	班级	姓名	学院	……

-- 图

G (V ,E)：结点集合、边集合

V：v1、v2、……

E：e1、e2、……



-- 数据的逻辑结构

-- 数据的存储结构

数 据 结 构

3 问题一、二涉及的问题

问题一、二涉及

- 如何描述数据、以及数据域数据之间的关系?

-- 数据的逻辑结构：集合、线性结构、树结构、图结构

-- 数据的存储结构：顺序结构、链接结构

→ **数 据 结 构**

- 如何解决实际问题?

-- 求解问题的方法：排序，查找（最短路径，最小元素）
-- 方法的有效性

→ **算法**

2 基本概念和术语

数据(Data)： 是对客观事物的符号表示，在计算机科学中是指所有能输入到计算机中并被计算机程序处理的符号的总称。对计算机科学而言，数据的含义极为广泛，如图象、声音等都可以通过编码而归之于数据的范畴。

数据元素(Data Element)： 是数据的基本单位，在计算机程序中通常作为一个整体进行考虑和处理。有时，一个数据元素可由若干个数据项(Data Item)组成。

数据项： 是数据的不可分割的最小单位。

数据对象(Data Object)： 是性质相同的数据元素的集合，是数据的一个子集。

2 基本概念和术语

数据结构(Data Structure)：是相互之间存在一种或多种特定关系的数据元素的集合。数据元素相互之间的关系称为结构(structure)。根据数据元素之间关系的不同特性，通常有下列四类基本结构：集合、线性结构、树形结构、图状结构或网状结构。

逻辑结构：描述的是数据元素之间的逻辑关系。

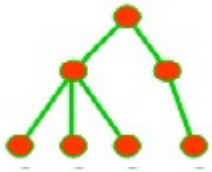
存储结构：数据结构在计算机中的表示(又称映象)称为数据的物理结构。

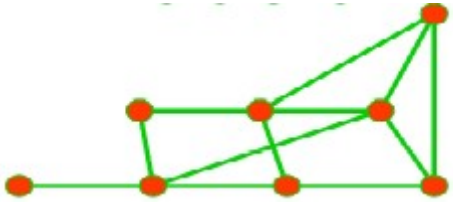
2 基本概念和术语

- 按照逻辑结构分类

集合  无关系

线性结构  一对一关系

树结构  一对多关系

图结构  多对多关系

3 算法

- 算法的五个特征

  -- 输入：初始值（零个或多个）

  -- 输出： 结果值（一个或多个）

  -- 有穷性

    - 执行有穷步后结束
    - 每一步都在有穷时间内完成

  -- 确定性

    - 每一条指令必须有确切的含义
    - 对于相同的输入只能得到相同的输出

  -- 可行性

    - 精确地运行、有限次运算实现

3 算法

问题的提出

- 求120、14 的最大公约数
- 求1234500、126的最大公约数

- 解决的办法

  -- 什么办法？同一个问题可以设计出不同的解决方法。

  -- 有否更好些的办法？如何评价？

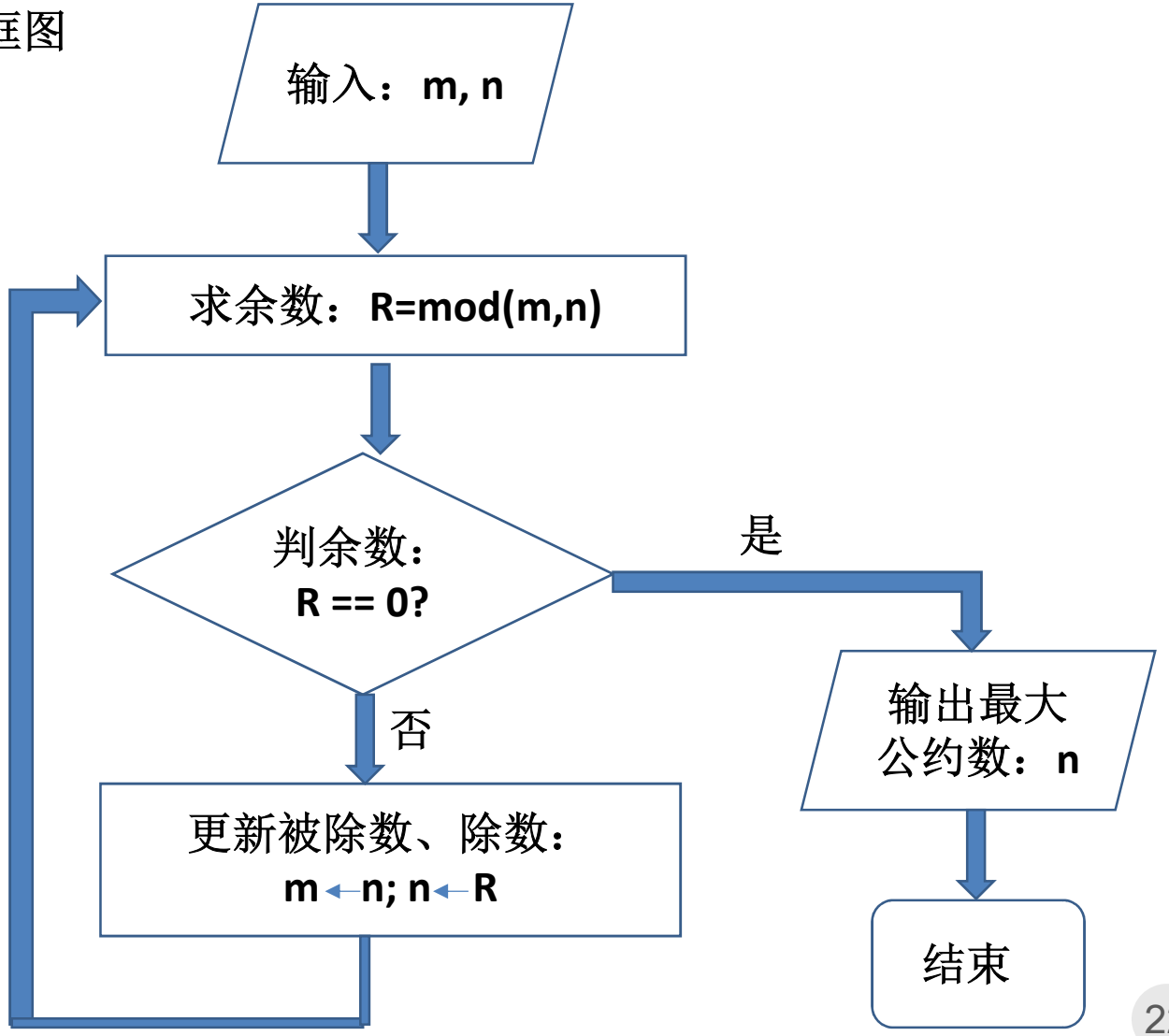
3 算法

算法：有穷规则的集合，其中的规则规定了一个解决某一个特定问题的运算序列。

-- 例：求18、12的最大公约数

3 算法

- 辗转相除法框图



4 时间复杂性的度量

- **算法的时间复杂性和空间复杂性**

-- 问题的规模： 或称大小，即为 n. 如：矩阵的阶数、图的结点个数

-- 时间复杂性： 算法所需的时间和问题规模的函数，记为 T (n)

  - 通过统计算法中基本操作重复执行的次数可近似地得到算法的执行效率，用O(n)表示，称为时间复杂度。

-- 空间复杂性： 算法所需的空间和问题规模的函数，记为 S (n)

4 时间复杂性的度量

-- 时间复杂性的评价方法

- 最坏情况下的时间复杂性（极端情况），如

  -- 把按递增排序的数组变为按递减排序

- 平均情况下的时间复杂性（等概率情况）

4 时间复杂性的度量

- **大 O 表示法**

-- 定义

- 如果存在一正的常数 C 和自然数 n_0
- 当 n>= n_0 时，有 T (n)<=C * F (n)成立
- 则称 T(n)=O(F(n))

-- 读法

- 如果一个算法的时间复杂性是O(g(n))，读作 g(n) “级”的或“阶”的
- 如：线性阶的、平方阶的、立方阶的……

- **大O表示法给出了算法在问题规模n达到一定程度后运行时间增长率的上界，因此被称为渐进时间复杂度，简称为时间复杂度。**

4 时间复杂性的度量

-- 例1、

设 $T(n) = (n+1)^2$

$= n^2 + 2n + 1$

$\leqslant n^2 + 2n^2 + n^2$

当 $n=1$ 时，等式成立

$n>1$ 时，$<$ 式成立

选 $n_0 = 1$，$C=4$，$T(n) \leqslant 4n^2$

所以，$T(n) = O(n^2)$

4 时间复杂性的度量

-- 注意（一）

- 通常所说的找到了时间复杂性的级别，是指找到了同样级别的最简单的函数

- 如：

-- 307n^2 、n^2 /2 、n^2 都是同一级别的函数，最简单的函数是 n^2

-- 307n^2 、n^2 /2 、n^2 的级别都是O(n^2)

4 时间复杂性的度量

-- 注意（二）

- $f(n)=O(g(n))$ 意味着找到了 $f(n)$ 的一个最“紧贴”的上界 $g(n)$。或者说找到了最低的上界。从算法的时间复杂性角度来看，下例中的$O(n^4)$是没有意义的

- 如：

-- $3n^3+2n^2$ 、 $2n^3$、n^5 、n^4中最紧贴上界为 $2n^3$

-- 最简单上界为 n^3

4 时间复杂性的度量

-- 求和定理

- 假设$T_1(n)$、 $T_2(n)$是程序P_1、 P_2 的运行时间，并且$T_1(n)$ 是$O(f(n))$的，而$T_2(n)$ 是$O(g(n))$的
- 那么，先运行P_1，再运行 P_2的总的运行时间是：

$$T_1(n)+ T_2(n)=O(MAX(f(n), g(n)))$$

-- 求积定理

- 如果$T_1(n)$和$T_2(n)$ 分别是$O(f(n))$ 和 $O(g(n))$的
- 那么，$T_1(n) * T_2(n)=O(f(n)*g(n))$