



第一章 程序设计基础



第一章 程序设计基础

- 了解：
 - 计算机体系结构
 - 程序流程图
 - 基本概念和历史等
- 熟练掌握
 - 标识符定义规则
 - 关键字(32个)
 - 常量和变量
 - main define include 虽然频繁出现，但不是关键字

第二章 基本数据类型及其运算

■ 基本数据类型(3种)

- 变量的4要素、整数的编码和存储、整数的进制和书面表示；
- 字符型的加减运算
- 实型的精度问题

■ 输入和输出

- 基本原则：转义字符和I/O数据的类型匹配。%f—float
- Scanf 用地址（什么时候用取地址符，格式匹配）
- 常用的转义字符
- cin和cout不要求

第二章 基本数据类型及其运算（续）

■ 运算

- 优先级会贯穿整个课程
- `++`、`--`、算术、**关系**、**逻辑**、条件、逗号、位运算
- 通过典型例子来记忆
- 考试前可以快速看优先级表格(强化记忆)

■ 表达式

- 关注整型数据的除法
- 强制类型转换和隐式类型转换

第三章 结构化程序设计

■ 基本语句

- 表达式语句、空语句、break、continue、return、goto
- 三种控制结构（顺序、分支、循环）
- 循环的四要素（初始化、循环条件、循环变量的改变、循环体）
- 写下思路，逐步求精

■ 注意

- break和continue的混合使用
- 分支结构中的关系判断（=和==）
- switch的要点（整型表达式、break、default）
- 空语句！！

第四章 数组

- 基本概念：定义、初始化、存储、使用
- 查找(有序和无序)、排序
- 二维数组的访问方式：顺序、顺时针、逆时针(作业)、斜向(4. 14)
- 字符数组和若干函数（注意如何使用存储字符串的二维数组）

■ 注意

- 越界问题
- 没有直接作用于整个数组的操作符
- 初始化和存储
- 字符串处理函数的使用限制

第五章 函数

- 基本概念：定义、声明
- 函数的要素（函数名、函数体、返回值、参数表）
- 递归函数(列递归方程，然后求解)
- 函数的执行顺序（包括递归函数）
- 存储类别和作用域(经常有3-5分的固定题目)
- 宏和预编译(经常有3-5分的固定题目)：替换原则，替换后再说
 - P111

■ 注意

- 参数类型匹配（若传入非匹配的数据，系统会进行隐式转换）
- static的存储类别和说明
- 函数的执行顺序(读程序…)
- 宏…

第六章 指针

- 指针的基本概念、
- 指向数组元素的指针（复制和查找）
- 指针作为形参 vs. 数组形参
- 指向二维数组一整行的指针 vs. 指针数组
- 多级指针和函数指针（了解）
- 引用（不要求）

■ 注意

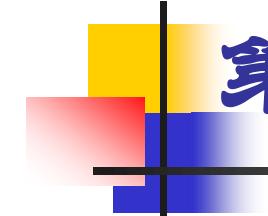
- 指针变量的存储空间只和系统配置等有关
- 深刻理解指针

第七章 结构

- 结构类型和结构变量
- 结构数组
- 结构和函数
- 链表：6个基本操作、表头和表尾指针、画图辅助思考
- 类型定义（3-5 如何定义数组）
- 链表的类型：普通单链表、双向链表、带辅助表头的链表、…
 - 6类基本操作能否移植到双向链表等上？
 - 查找、插入、删除、更新、合并、旋转…

■ 注意

- 形式多样 所以（深刻理解基本问题、并学会如何使用图示）
- 类型定义



第八章 文件

- 文件指针如何定义
 - 常用库函数的使用(关注参数 fopen-r)
 - 读和改错 例8.8
-
- 注意
 - 注意库函数的使用
 - 可以快速阅读有关文件操作的代码

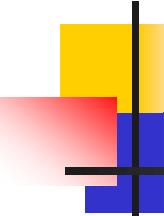
陷阱凶没

- 整型数据的除法
- Scanf的取地址符
- I/O中的%f—double
- 字符的加减 ‘b’ = ‘a’ +1;
- =和== 的陷阱 if (a==b)
- 逻辑运算的运算规则
- ++和--的含义
- break和continue用错
- 无处不在的空语句 (经常和宏搭配使用)
- 指针的初始化
- 宏 (大多宏出现的地方，都要有巨大危机感)
- 数组访问的越界
- 循环变量 (某些统计值) 的初始化



陷阱凶没（续）

- 递归函数的退出条件
- 函数的指针形参
- 数组的初始化
- 指针指向错误（要结合逻辑排查）
- 文件的库函数使用
- 字符数组相关函数
- ...



复习要点

- 书本
 - 每章开篇有目标、结束有小结
 - 例题和课后作业（读和想）
- 上课的课件
- 上机指导（习题）