

《C 程序设计》期末复习知识点提示

第 1 章

了解常用进制的表示(表 1-1)和相互转换。

整型二进制数的位运算包括按位取反~、按位与&、按位或|、按位异或^、左移<<、右移>>共 6 种

第 2 章

从机器语言、汇编语言到高级语言的发展方向是方便人和计算机交流信息。

C 语言是 Thompson 和 Ritchie 在上世纪七十年代初在前人基础上发明的。

三个程序(例 2.1、例 2.2、例 2.3)举例及相应的解释。

编制运行 C 程序基本步骤: (1)编辑源程序; (2)编译成目标文件; (3)链接成可执行程序;

第 3 章

程序=算法+数据结构

标识符规则: 只能由字母、数字、下划线组成, 且第一个字母必须是字母或下划线(不允许使用保留关键字见附录, 如 int, char, if, else, switch, while 等)。适用于变量名、函数名、数组名、指针变量名, 复杂数据类型名等的命名

数据分类: 按表现形式分常量、变量; 按数据类型分整型、字符型、单双精度实型等;

常量:

1) 整型常量: 八进制 015, 0101, -012; 十六进制 0X2A, -0xA0; 长整型 123L, 45l, 0XAFL; 无符号数 358U, 0x38Au, 235Lu, 0XA5Lu

2) 实型常量(双精度): .25, 2.1E5, -3.7e-2, 1.34e3, -0.145E-9

3) 字符常量表示和存储(ASCII 码): 字母'c','C', 空格' ', 数字'0'~'9','1', '7'(注意和整型常量 7 的区别), 控制或转义字符'\0', '\n', '\t', '\a'

4) 字符串常量: 双引号“ ”, 末尾必有'\0'结尾。用户自定义字符串数组初始化如 char str[20]={“China”}; char str[]=”China”; 字符串常量的起始地址可以赋值给指针变量 char *cp=”China”; 注意二者的区别。

5) 符号常量: 符号常量名一般大写, 末尾不能有分号。#define NUM 20

变量: 常用的基本数据类型有 int, char, float, double, 确定给变量分配存储单元的数量(可用 sizeof(类型名)取得)以及如何存储数据, 从而确定了数据的表示范围、精度。

根据题目要求和使用需求定义合适的变量类型 int, char, float, double

(1)整型变量 int:

(2)字符变量 char: %c 输出, 保存的是对应的 ASCII 码%d, 本质上是一个字节的整型变量。字符之间、字符和整型之间可运算'a'+4

(3)单精度浮点型变量 float:

(4)双精度浮点型变量 double:

(5)类型不同时的自动类型转换如 int a=3.14;后, a=?

(6)不同类型间混合运算自动向高级别数据类型转换;

(7)强制类型转换 (int)x+y, (int)(x+y), (int *)malloc(N*sizeof(int))

复合赋值运算符+=, -=, *=, /=:

算术运算符: /除法, 两个整数相除还是整数, 必考题(3/5=?); %整数求余; fabs(x)实数取绝对值

自增自减运算符: 只能用于变量, 不能用于常量或表达式

(1)++i, --i: 先自增自减, 再使用 (即输出、参与运算等)

(2)i++, i--: 先使用 (即输出、参与运算等), 再自增自减

(3)用于数组元素或结构体成员 a[i]++, p+i->age++

(4)用于指向数组的指针变量, 依次指向下一个元素 int* p=a+3; p++

逗号表达式: 会依次计算各个表达式的值, 并将最后一个表达式的值作为整个逗号表达式的值 d = (a=3, b=4, 4*5), d=?

位运算符, 位运算表达式的计算

第 4 章

printf, scanf 函数: 非常重要

printf(“a= %-12.5f \n”, add(123.456, x)); printf(“i= %-5d\n”, i*i+9);

scanf()参数是地址列表, 普通变量千万别忘记取地址符号&, 并且输入数据时一定要按照格式占位符规定的形式输入。

整型	int i,j; printf("Input 2 integers: "); scanf("%d%d", &i,&j); //数据输入用空格或回车分隔 scanf("%d,%d", &i,&j); //数据输入时必须用逗号分隔	printf("integer: %d\n", i);
字符型	char ch1,ch2; scanf("%c%c", &ch1, &ch2); //注意: 每一个键盘输入的字符都算, 空格、换行回车符也是普通字符也会被输入	printf("char: %c %d\n", ch,ch); //既可以字符也可以ASCII 码整数输出
单精度	float x,y; scanf("%f %f", &x,&y); //数据输入用空格或回车分隔	printf("result: %f\n", x);
双精度	double y; scanf("%lf", &y); //双精度格式占位符一定要用%lf	printf("value = %f\n", y); printf("value = %g\n", y);
字符串	char str[20]; scanf("%s", str); //不加& //遇到空格时字符串接收会结束, 最好用 gets(str)	printf("%s", str); //也可以用 puts(str);

printf()特殊字符的输出: 回车换行\n, 制表符\t, 双引号\", 反斜线\\, 百分号%, 输出警告声音\a, 输出八进制对应字符\ooo

单个字符输入 ch=getchar(); 单个字符输出 putchar(ch); putchar(‘a’); putchar(‘\n’);

能看懂流程图表示的算法（考试不要求画），平时使用伪代码表示算法
结构化编程只需用三种基本结构即可完成：顺序结构、选择结构、循环结构（当型循环，直到型循环）。

三种基本结构特点：(1) 只有一个入口；(2) 只有一个出口；(3) 结构内的每一部分都有机会被执行到；(4) 结构内不存在“死循环”；

重点算法掌握：累加和、累乘积、打擂台找最大最小值(下标)、冒泡法排序、选择法排序

第 5 章 选择结构

关系运算符共 6 种，判断相等“==”一般只用于整型或字符型，实型用 `fabs(x-a)<EPS` 来判断
逻辑运算符&&, ||, !。如何判断闰年？
条件运算符: `max=a>b?a:b;`
if [else if, else]语句:
switch case 语句:

第 6 章 循环结构

三种循环语句，重点是 for 语句
while 和 do while 语句的区别
break 和 continue 语句的作用和区别

第 7 章 利用数组处理批量数据

数组的定义，数组元素的引用
数组的初始化
利用 **for** 语句遍历整个数组元素的引用
字符串数组，字符串处理函数
算法：打擂台找最大最小值、冒泡法排序、选择法排序

第 8 章

子函数的定义（声明）、调用、参数和返回值。
利用数组名作为参数向自定义函数传递一维数组的首地址
局部变量和全局变量。它们作用域(空间角度)和生存期(时间角度)。

第 9 章 指针

务必理解指针的概念，保存的是地址，间接存取；指针的定义和初始化
地址运算符&, 指针运算符*
指针变量作为函数的参数
一维数组的地址和指针引用数组元素
指针和字符串
输入输出整个数组的常用写法

数组元素地址写法		数组元素引用写法	
a	a	a[0]	*a

<code>&a[0]</code>	<code>p=a</code>	<code>p[0]</code>	<code>*p</code>
<code>&a[i]</code>	<code>a+i</code>	<code>a[i]</code>	<code>*(a+i)</code>
<code>&p[i]</code>	<code>p+i</code>	<code>p[i]</code>	<code>*(p+i)</code>

整型或实型数组	字符串数组
<pre>int i, a[10]={9,8}, *p=a; for(i=0; i<10; i++) printf("%d", a[i]); printf("%d", p[i]); printf("%d", *(a+i)); printf("%d", *(p+i)); printf("%d", *p++); for(p=a; p<a+10; p++) printf("%d", *p);</pre>	<pre>int i, str[20]="China", *p=str; for(i=0; str[i]!='\0'; i++) str[i] = 'A' + i; *(str + i) = 'A' + i; for(p=str; *p!='\0';; p++) *p = 'A' + (p-str);</pre>

动态内存分配函数 malloc，free

第 10 章
不带参数的宏定义

第 11 章
结构体类型的定义，结构体变量的定义，结构体变量初始化；
指向结构体变量的指针；
结构体成员的两种引用方式：变量名使用成员运算符 `stu.num`，指针使用指向运算符 `p->num`
结构体变量数组
共用体类型的定义，变量的定义，共用体各成员共同占有同一段内存单元，其大小由占据最大内存空间成员所占的空间数决定。

第 12 章
文件指针 `FILE *fp;`
文件的打开 `fopen`，特别注意 `fopen` 的文件各种打开方式的区别，如“r”，“w”，“r+”，“w+”，“a+”的区别等。默认以文本格式打开。
了解文本文件格式化读写函数 `fscanf`，`fprintf`

《C 程序设计》超出教学大纲基本要求的内容

对应教材《C 语言程序设计教程》(第 3 版)，王敬华主编，清华大学出版社，2021

- 6.1.6 goto 语句
- memset、memcpy 函数

- 3) 8.5 变量的存储类型
- 4) 8.7 函数的作用域
- 5) 8.9 函数封装
- 6) 8.10 函数应用综合举例
- 7) 9.7 多级指针
- 8) 9.10 指向函数的指针——函数指针
- 9) 9.11 带参数的 `main` 函数
- 10) 10.2.2 带参数的宏定义
- 11) 10.4 条件编译
- 12) 11.2.4 结构体变量内存分配问题透析
- 13) 11.3 线性链表
- 14) 11.4 联合体
- 15) 11.5 位域
- 16) 11.6 枚举类型变量的定义和引用
- 17) 11.7 复杂数据类型应用综合举例
- 18) 12.6 文件的定位读写
- 19) 12.7 文件应用综合举例