

C语言程序设计(第7章结构体与共用体) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/135/2021_2022_C_E8_AF_AD_E8_A8_80_E7_A8_8B_c97_135095.htm 7.1 结构体类型变量的定义和引用在上面描述的各种登记表中，让我们仔细观察一下住宿表、成绩表、通讯地址等。住宿表由下面的项目构成：这些登记表用C提供的结构体类型描述如下：住宿表:struct accommod{ char name[20]. /*姓名*/ char sex. /*性别*/ char job[40]. /*职业*/ int age. /*年龄*/ long number. /*身份证号码*/}. 成绩表:struct score{ char grade[20]. /* 班级*/ long number. /* 学号*/ char name[20]. /*姓名*/ float os. /*操作系统*/ float datastru. /* 数据结构*/ float compnet. /* 计算机网络*/}. 通讯地址表:struct addr{ char name[20]. char department[30]./* 部门*/ char address[30]. /*住址*/ long box. /* 邮编*/ long phone. /* 电话号码*/ char email[30]. /* Email */}; 这一系列对不同登记表的数据结构的描述类型称为结构体类型。由于不同的问题有不同的数据成员，也就是说有不同描述的结构体类型。我们也可以理解为结构体类型根据所针对的问题其成员是不同的，可以有任意多的结构体类型描述。下面给出C对结构体类型的定义形式：struct 结构体名{ 成员项表列；}; 有了结构体类型，我们就可以定义结构体类型变量，以对不同变量的各成员进行引用。7.1.1 结构体类型变量的定义 结构体类型变量的定义与其它类型的变量的定义是一样的，但由于结构体类型需要针对问题事先自行定义，所以结构体类型变量的定义形式就增加了灵活性，共计有三种形式，分别介绍如下：1) 先定义结构体类型，再定义结构体类型变量：struct stu /*定义学

生结构体类型* /{ char name[20]. /* 学生姓名*/ char sex. /* 性别*/ long num. /*学号*/ float score[3]. /* 三科考试成绩*/}.struct stu student1,student2./* 定义结构体类型变量*/struct stu student3,student4. 用此结构体类型，可以定义更多的该结构体类型变量。2) 定义结构体类型同时定义结构体类型变量

: struct data{ int day. int month. int year.} time1,time2. 也可以再定义如下变量：struct data time3,time4. 用此结构体类型，同样可以定义更多的该结构体类型变量。3) 直接定义结构体类型变量：struct{ char name[20]. /*学生姓名*/ char sex. /*性别*/ long num. /*学号*/ float score[3]. /*三科考试成绩*/}

person1,person2. /*定义该结构体类型变量*/ 法由于无法记录该结构体类型，所以除直接定义外，不能再定义该结构体类型变量。7.1.2 结构体类型变量的引用 学习了怎样定义结构体类型和结构体类型变量，怎样正确地引用该结构体类型变量的成员呢？C 规定引用的形式为：类型变量名>. 若我们定义的结构体类型及变量如下：struct data{ int day. int month. int year.} time1,time2. 则变量time1和time2各成员的引用形式为：time1.day、time1.month、time1.year及time2.day

、time2.month、time2.yea r，如图7 - 1所示。类型变量的各成员与相应的简单类型变量使用方法完全相同。7.1.3 结构体类型变量的初始化 由于结构体类型变量汇集了各类不同数据类型的成员，所以结构体类型变量的初始化就略显复杂。结构体类型变量的定义和初始化为：struct stu /* 定义学生结构体类型*/ /{ char name[20]. /* 学生姓名*/ char sex. /* 性别*/ long num. /*学号*/ float score[3]. /* 三科考试成绩*/}.struct stu student={"liping",f,970541,98.5,97.4,95}. 上述对结构体类型变量

的三种定义形式均可在定义时初始化。结构体类型变量完成初始化后，即各成员的值分别为：`student.name = "liping"`、`student.sex = f`、`student.num = 970541`、`student.score[0] = 98.5`、`student.score[1] = 97.4`、`student.score[2] = 95`。其存储在内存的情况如图7 - 2所示。我们也可以通过C提供的输入输出函数完成对结构体类型变量成员的输入输出。由于结构体类型变量成员的数据类型通常是不一样的，所以要将结构体类型变量成员以字符串的形式输入，利用C的类型转换函数将其转换为所需类型。类型转换的函数是：`int atoi(char *str)`；转换str所指向的字符串为整型，其函数的返回值为整型。`double atof(char *str)`；转换str所指向的字符串为实型，其函数的返回值为双精度的实型。`long atol(char *str)`；转换str所指向的字符串为长整型，其函数的返回值为长整型。使用上述函数，要包含头文件"stdlib.h"。对上述的结构体类型变量成员输入采用的一般形式：`char temp[20].gets(student.name). /* 输入姓名*/student.sex = getchar(). /* 输入性别*/gets(temp). /*输入学号*/student.num = atol(temp). /* 转换为长整型*/for(i = 0. i { gets(temp). student.score[i] = atoi(temp).}`对该结构体类型变量成员的输出也必须采用各成员独立输出，而不能将结构体类型变量以整体的形式输入输出。C允许针对具体问题定义各种各样的结构体类型，甚至是嵌套的结构体类型。`struct data{ int day. int month. int year.}.struct stu{ char name[20]. struct data birthday. /* 出生年月，嵌套的结构体类型*/ long num.} person.`该结构体类型变量成员的引用形式：`person.name`、`person.birthday.day`、`person.birthday.month`、`person.birthday.year`、`person.num`。100Test 下载频道开通，各类考试

题目直接下载。详细请访问 www.100test.com