C语言程序设计基础

林川

第9章 结构

- 什么是结构？
- 定义和使用
- 结构+数组
- 结构+指针
- 结构+函数

9.1 结构定义和使用

- 数据类型
  - C语言提供的基本数据 类型
    - 整数：int，unsigned，short，long，……
    - 浮点数：float，double
    - 字符：char
      - 字符串不是基本的数据 类型
    - 空/无类型：void
    - 指针：各种数据 类型都有对应的指针类型
    - 数组：各种数据 类型都有对应的数组类型
      - 不同长度、不同维度的数组是不同的类型

9.1 结构定义和使用

- 结构类型
  - 用户自定义的数据类型

struct是C语言关键字
student是用户定义的标识符，作为结构的名字，必须和struct联合使用。

```
struct student
{
    int  num;                          /*学号*/
    char name[10]                      /*姓名*/
    int  computer, english, math;      /*成绩*/
    double average;                    /*平均成绩*/
}; /* 不要遗漏分号 */
```

9.1 结构定义和使用

```
struct 结构名
{
        类型名 结构成员名1;
        类型名 结构成员名2;
        . . .
        类型名 结构成员名n;
};
```

- 关键字struct和结构名一起，构成一个数据类型
- 结构的定义以分号结束，被看作一条语句(结构定义语句)
- 一个结构体所占的字节数可以sizeof运算符确定

结构定义示例

定义平面坐标结构：

```
struct point
{
    double  x;
    double  y;
};
```

或者

```
struct point
{
    double  x, y;
};
```

定义一个图像

```
struct image
{
   int  width, height;
   int  format;
   char * pixels;
};
```

定义一个产品

```
struct prooduct
{
   int  id;
   int  type;
   char name[100];
   int  price;
};
```

结构定义示例

定义一个复数:

```
struct complex
{
    double  real, image;
};
```

定义一个地址

```
struct address
{
   char city[20];
   char street[20];
   char code;
   int  zip;
};
```

定义一个朋友

```
struct friend
{
    char name[10];
    char phone[13];
    int  age;
    struct address addr;
    char memo[100];
};
```

9.1 结构定义和使用

[例9-1] 建立一个学生信息库

```
struct student
{
    int  num;                          /*学号*/
    char name[10]                      /*姓名*/
    int  computer, english, math;      /*成绩*/
    double average;                    /*平均成绩*/
};
```

[例9-1] 建立一个学生信息库

```
#define MaxSize 50
struct student students[MaxSize];
int count = 0;
int a[10];
```

用结构struct student
定义了一个数组students
长度为50

MaxSize是一个宏，定义为50

宏定义的一般格式
#define 宏名 宏体

之后所有的宏名都会被编译器替换为宏体

[例9-1] 建立一个学生信息库

```
void new_student(struct student students[])
{
   struct student s;

   if( count==MaxSize ) {
     printf("The array is full\n");
     return;
   }
   scanf("%d", &s.num);
   scanf("%s", &s.name);
   scanf("%d", &s.math);
   scanf("%d", &s.computer);
   scanf("%d", &s.english);
   s.verage = ( s.math + s.computer + s.english ) / 3.0;
   students[count ++] = s;
}
```

结构数组/指针作为形式参数
等价于struct student *students

用运算符 .
使用结构成员变量

结构变量可以整体赋值

[例9-1] 建立一个学生信息库

```
/* 输出一个学生的信息 */
void print_student(struct student s)
{
   printf("Num: %d", s.num);
   printf("Name: %s", s.name);
   printf("Math: %d", s.math);
   printf("Computer: %d", s.computer);
   printf("English: %d", s.english);
   printf("Average: %.2f\n", s.average);
}
```

结构类型的形式参数

[例9-1] 建立一个学生信息库

```
/* 根据学号查找学生 */
void search_student(struct student students[],
                    int num )
{
   int i;
   for( i=0; i<count; i++ )
       if( students[i].num == num )
       {
          print_student( students[i] );
          return;
       }
   printf("Not Found\n");
}
```

结构数组的第i个元素
作为实际参数

[例9-1] 建立一个学生信息库

```
/* 根据学好查找学生 */
void output_student(struct student students[])
{
   int i;
   if( count==0 )
   {
      printf("Count of student is zero\n");
      return;
   }

   for( i=0; i<count; i++ )
      print_student( students[i] );
}
```

9.2.2 结构变量的定义和初始化

1 单独定义

```
struct student
{
    int  num;                          /*学号*/
    char name[10]                      /*姓名*/
    int  computer, english, math;      /*成绩*/
    double average;                    /*平均成绩*/
};
```

```
struct student s1;
```

数据类型是struct student
结构变量是s1

9.2.2 结构变量的定义和初始化

2 混合定义

```
struct student
{
    int  num;                   /*学号*/
    char name[10]               /*姓名*/
    int  computer, english, math; /*成绩*/
    double average;             /*平均成绩*/
} s1, s2 ;
```

数据类型是struct student

结构变量是s1, s2

9.2.2 结构变量的定义和初始化

3 无名定义

```
struct
{
    int  num;                          /*学号*/
    char name[10]                      /*姓名*/
    int  computer, english, math;      /*成绩*/
    double average;                    /*平均成绩*/
} s1, s2 ;
```

数据类型是一种结构类型，但是没有给它名字

结构变量是s1，s2

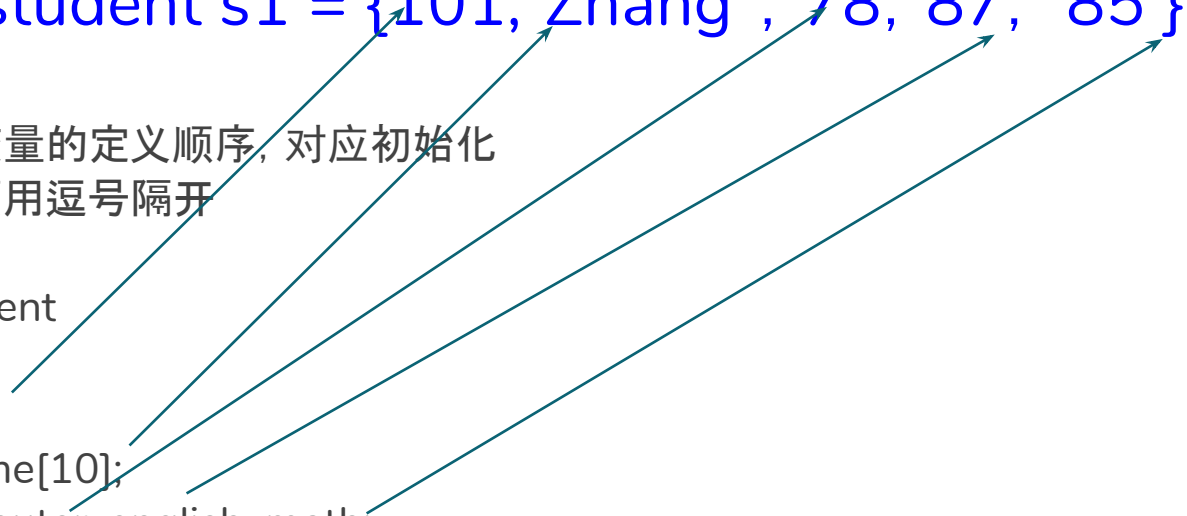
9.2.2 结构变量的定义和初始化

结构变量初始化

struct student s1 = {101,“Zhang”, 78, 87, 85 };

按照成员变量的定义顺序，对应初始化
各个数据项用逗号隔开

```
struct student
{
    int  num;
    char name[10];
    int  computer, english, math;
    double average;
};
```



9.2.2 结构的使用

1 引用结构成员

结构变量名.结构成员名

```
s1.num = 101;
strcpy(s1.name, “zhang”);
s2.num = s1.num;
strcpy(s2.name, s1.name);
```

2 结构整体赋值

```
s2 = s1;
```

相当于将s2中所有字节中内容复制到s1中

9.2.2 结构的使用

3 定义结构数组

struct student students[50];

4 结构类型的参数

void print_student(struct student s);

5 结构变量作为返回值

当程序的规模较大，功能较多时，需要以函数的形式进行功能模块的划分和实现；

如果在函数间传递结构数据，则需用结构变量作为函数的参数或返回值。

9.3 结构数组

- 结构数组的定义方法与结构变量相同

```
struct friends_list
{
      char name[10];
      int age;
      char telephone[13];
}   friends[10];
```

  - 定义了结构数组friends
  - 它有10个数组元素，从friends[0]到friends[9]
  - 每个数组元素都是结构类型struct friends_list

结构数组的初始化

```
struct friends_list friends[10] =
{
    { "zhang san", 26, "0571-85271880"},
    { "Li Si", 30, "13605732436"}
};
```

- 初始化了2个元素：friends[0]，friends[1]
- 规则：数组初始化规则+结构初始化规则

9.3 结构数组

- 一个结构变量只能表示一个实体的信息，如果有许多相同类型的实体，就需要使用结构数组。
- 结构数组是结构与数组的结合，与普通数组的不同之处在于每个数组元素都是一个结构类型的数据，包括各个成员项。

[例9-3] 结构数组排序

```
struct student sa[50];
int n, i ;

/* 输入一批学生和成绩 */
scanf(“%d”, &n);
for( i=0; i<n; i++ )
{
   scanf(“%d”, &sa[i].num);
   scanf(“%s”, &sa[i].name);
   ……/*成绩略*/
   as[i].average = …
}
```

```
struct student
{
    int  num;
    char name[10];
    int computer, english, math;
    double average;
};
```

9.3 结构数组

```
/* 按照平均分对学生数组排序 */
void sort(struct student s[], int n)
{
   int i, j, index;
   struct student temp;
   for( i=0; i<n-1; i++ )
   {
      index = i;
      for( j=i+1; j<n; j++ )
         if( s[j].average>s[index].average )
             index = j;
      temp = s[index];
      s[index] = s[i];
      s[i] = temp;
   }
}
```

结果是从高到低

9.4 结构指针

结构指针

指向结构类型变量的指针

```
struct student sa[50], s1, s2, *p;
struct student *p2 = &s2;
p = &s1;
p = &s2;
p = sa + 5; p = &sa[5];
```

9.4 结构指针

使用结构指针

```
struct student s1, s2, *p = &s1;

(*p).num = 101;
```

或者

```
p->num = 101;
```

- -> 箭头运算符（减号+大于号）
- 访问指针所指向的结构的成员
- 优先级非常高，和点运算符是一样.

本章要点

- 什么是结构？
- 定义形式
- 结构嵌套
- 结构变量和结构成员变量，
- 引用结构成员变量
- 结构在函数参数中使用
- 结构数组，如何定义和使用结构数组？
- 结构指针，通过结构指针访问结构成员