

C语言程序设计

计算机科学与技术学院

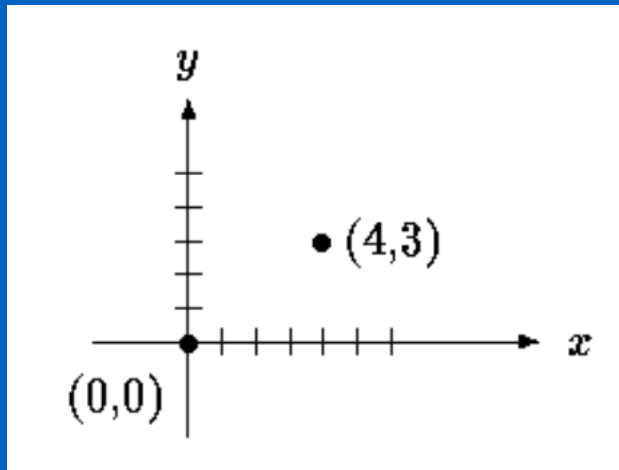
结构体

- 结构体和结构体变量
- 结构体变量的声明、定义和引用
- 结构体数组
- 结构体的应用



数据如何存储？

- 一条信息由多个数据才能描述



- 一个点的坐标：

```
int x;
```

```
int y;
```

```
x = 4;
```

```
y = 3;
```

- 10个点？ 100000个点？
- 如何反映同一个点的两个坐标之间存在的内在联系？

数据如何存储？

- 有学生成绩表如下：

| 学号 | 姓名 | 课程 | 学分 | 成绩 | 获得绩点 |
|------|----|----------|-----|----|------|
| 1001 | 李芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 1001 | 李芳 | 高等数学 (1) | 6.0 | 80 | 3.0 |
| 1001 | 李芳 | 大学英语 (1) | 4.0 | 75 | 2.5 |
| 1002 | 赵力 | C语言程序设计 | 4.0 | 90 | 4.0 |
| 1002 | 赵力 | 高等数学 (1) | 6.0 | 85 | 3.5 |
| 1002 | 赵力 | 大学英语 (1) | 4.0 | 80 | 3.0 |
| 1003 | 王倩 | C语言程序设计 | 4.0 | 60 | 1.0 |
| 1003 | 王倩 | 高等数学 (1) | 6.0 | 70 | 2.0 |
| 1003 | 王倩 | 大学英语 (1) | 4.0 | 80 | 3.0 |

- 一个学生：

```
int    num;  
char   name[20];  
char   course_name[20];  
float  credit;  
int     grade;  
float  GP;
```

- 10个学生？ 10000个学生？
- 如何反映同一个学生的多个数据之间存在的内在联系？

复合数据类型

- 复合数据类型：由多个基本数据类型复合而成
 - 单一数据 => 数据集合
- 数组
 - 相同类型的数据构成
 - 每个数据是一个元素(element)
 - 元素的名称是 数组名[下标]
- 结构体(structure)
 - 可由不同类型的数据构成
 - 每个数据是一个成员(member)
 - 成员的名称可定义

结构体：数据类型 和 变量

- 结构体(*structure*)

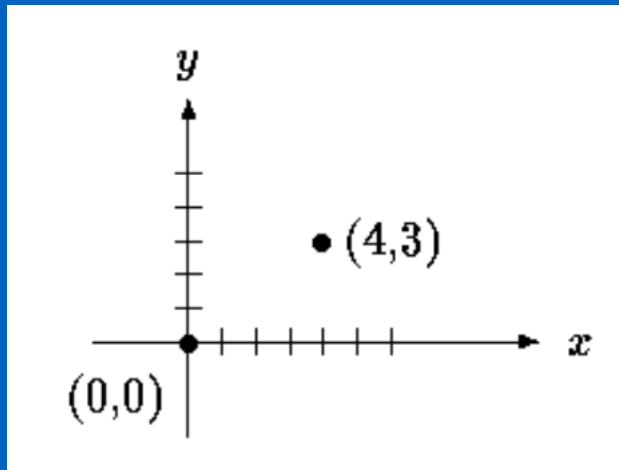
- 可由不同类型的数据构成
- 每个数据是一个成员(member)
- 成员的名称可定义

- 一般格式：

```
struct 结构体名 {  
    数据类型1 成员名1;  
    数据类型2 成员名2;  
    .....  
};  
struct 结构体名 结构变量 = {初始数据};
```

定义结构体数据类型 和 结构体变量

- 一条信息由多个数据才能描述



- 一个点的坐标:

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

```
struct point pt;
```

定义结构体数据类型 和 结构体变量

- 有学生成绩表如下：

| 学号 | 姓名 | 课程 | 学分 | 成绩 | 获得绩点 |
|------|----|----------|-----|----|------|
| 1001 | 李芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 1001 | 李芳 | 高等数学 (1) | 6.0 | 80 | 3.0 |
| 1001 | 李芳 | 大学英语 (1) | 4.0 | 75 | 2.5 |
| 1002 | 赵力 | C语言程序设计 | 4.0 | 90 | 4.0 |
| 1002 | 赵力 | 高等数学 (1) | 6.0 | 85 | 3.5 |
| 1002 | 赵力 | 大学英语 (1) | 4.0 | 80 | 3.0 |
| 1003 | 王倩 | C语言程序设计 | 4.0 | 60 | 1.0 |
| 1003 | 王倩 | 高等数学 (1) | 6.0 | 70 | 2.0 |
| 1003 | 王倩 | 大学英语 (1) | 4.0 | 80 | 3.0 |

```
//定义结构体数据类型
```

```
struct student {  
    int    num;  
    char   name[20];  
    char   course_name[20];  
    float  credit;  
    int    grade;  
    float  GP;  
}
```

```
//定义结构体变量
```

```
struct student stu1, stu2;
```

```
//定义结构体数据类型
```

```
struct student {  
    int    num;  
    char   name[20];  
    char   course_name[20];  
    float  credit;  
    int    grade;  
    float  GP;  
}
```

```
stu1, stu2; //和 变量
```


结构体类型与结构体变量概念不同

- 类型

- 是“模板”或“图纸”
- 不分配内存；
- 不能存储数据
- 不能赋值、存取、运算；

```
//定义结构体数据类型
struct point {
    int x;
    int y;
};

struct student {
    int num;
    char name[20];
    char course_name[20];
    float credit;
    int grade;
    float GP;
}
```

- 变量

- 是内存中的“实体”
- 占据一定的内存空间；
- 可以存储数据
- 可以赋值、存取、运算；

```
//定义结构体变量
struct point pt;
struct student stu1, stu2;
```

结构体变量：初始化

```
struct point {  
    int x;  
    int y;  
};  
struct point pt = {4, 3};
```

//定义结构体数据类型

```
struct student {  
    int num;  
    char name[20];  
    char course_name[20];  
    float credit;  
    int grade;  
    float GP;  
}
```

//定义结构体变量 并初始化

```
struct student stu1 = {  
    1001,  
    "李芳",  
    "C语言程序设计",  
    4.0,  
    85,  
    3.5  
};
```

- 变量初始化：成员的类型和次序一一对应
- 存储：4 + 20 + 20 + 4 + 4 + 4 = 56 字节

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

结构体变量： 引用成员

- 结构体变量不能整体引用
- 只能引用变量成员
 - 方式： 结构体变量名.成员名
 - .:成员(分量)运算符
 - 优先级: 1
 - 结合性:从左向右
- 结构变量的成员与普通变量一样

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

```
//定义结构体数据类型
struct student {
    int    num;
    char   name[20];
    char   course_name[20];
    float  credit;
    int    grade;
    float  GP;
}

//定义结构体变量 并初始化
struct student stu1 = {
    1001,
    "李芳",
    "C语言程序设计",
    4.0,
    85,
    3.5
};

printf(
    "%d\t%s\t%s\t%.1f\t%d\t%.1f\n",

    stu1.num,
    stu1.name,
    stu1.course_name,
    stu1.credit,
    stu1.grade,
    stu1.GP
);
```

结构体变量：引用成员

- 结构体变量不能整体引用

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

//定义结构体数据类型

```
struct student {  
    int    num;           //成员是整数  
    char   name[20];      //成员是数组  
    char   course_name[20]; //成员是数组  
    float  credit;        //成员是浮点数  
    int    grade;         //成员是整数  
    float  GP;            //成员是浮点数  
}  
struct student stu1;
```

```
stu1.num = 1001;  
//stu1.name="张芳";  出错  
strcpy(stu1.name, "李芳");  
strcpy(stu1.course_name, "C语言程序设计");  
stu1.credit = 4.0;  
stu1.grade = 85;  
stu1.GP = 3.5;  
  
printf(  
    "%d\t%s\t%s\t%.1f\t%d\t%.1f\n",  
  
    stu1.num,  
    stu1.name,  
    stu1.course_name,  
    stu1.credit,  
    stu1.grade,  
    stu1.GP  
);
```

结构体变量： 引用成员

- 结构变量的成员与普通变量一样

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

//定义结构体数据类型

```
struct student {  
    int    num;           //成员是整数  
    char   name[20];      //成员是数组  
    char   course_name[20]; //成员是数组  
    float  credit;        //成员是浮点数  
    int    grade;         //成员是整数  
    float  GP;            //成员是浮点数  
}
```

//定义结构体变量

```
struct student stu1;
```

```
printf("请输入一个同学的成绩记录： \n");
```

```
scanf("%d%s%s%f%d%f",  
      &stu1.num, //成员是整数类型  
      stu1.name, //成员数组名本身即地址  
      stu1.course_name, //同上  
      &stu1.credit,  
      &stu1.grade,  
      &stu1.GP  
      );
```

```
printf("%d\t%s\t%s\t%.1f\t%d\t%.1f\n",  
      stu1.num,  
      stu1.name,  
      stu1.course_name,  
      stu1.credit,  
      stu1.grade,  
      stu1.GP  
      );
```

结构体变量： 引用成员

- 结构变量的成员与普通变量一样

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

//定义结构体数据类型

```
struct student {  
    int    num;           //成员是整数  
    char   name[20];      //成员是数组  
    char   course_name[20]; //成员是数组  
    float  credit;        //成员是浮点数  
    int    grade;         //成员是整数  
    float  GP;            //成员是浮点数  
}
```

//定义结构体变量

```
struct student stu1;
```

```
printf("请输入一个同学的成绩记录： \n");
```

//scanf("%d",stu1.num);出错

```
scanf("%d",&stu1.num);
```

//scanf("%s",&stu1.name); 出错

```
scanf("%s",stu1.name);
```

```
gets(stu1.course_name);
```

```
scanf("%d",&stu.credit);
```

```
scanf("%d",&stu.grade);
```

```
scanf("%d",&stu.GP);
```

```
printf("%d\t%s\t%s\t%.1f\t%d\t%.1f\n",  
    stu1.num,  
    stu1.name,  
    stu1.course_name,  
    stu1.credit,  
    stu1.grade,  
    stu1.GP  
);
```

结构体变量：赋值

- 结构变量的成员与普通变量一样

| | | | | | |
|------|-----|---------|-----|----|-----|
| 1001 | 张芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 4B | 20B | 20B | 4B | 4B | 4B |

```
//定义结构体数据类型
struct student {
    int    num;           //成员是整数
    char   name[20];      //成员是数组
    char   course_name[20]; //成员是数组
    float  credit;        //成员是浮点数
    int    grade;         //成员是整数
    float  GP;            //成员是浮点数
} stu2;
```

```
//定义结构体变量
struct student stu1 = {
    1001,
    "李芳",
    "C语言程序设计",
    4.0,
    85,
    3.5
};
```

```
stu2 = stu1; //结构体赋值
```

```
printf("%d\t%s\t%s\t%.1f\t%d\t%.1f\n",
    stu2.num,
    stu2.name,
    stu2.course_name,
    stu2.credit,
    stu2.grade,
    stu2.GP
);
```

结构体变量

- 输入两个学生的学号、姓名和成绩
- 输出成绩较高学生的学号、姓名和成绩

```
10101 Wang 89
10103 Ling 90
The higher score is:
10103 Ling 90.00
```

```
#include <stdio.h>
//定义结构体数据类型
struct student {
    int    num;           //成员是整数
    char   name[20];      //成员是数组
    float  score;         //成员是浮点数
} student1, student2;
```

```
int main(){

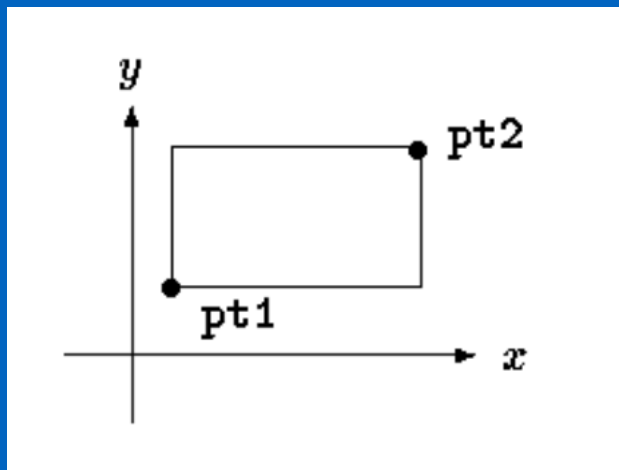
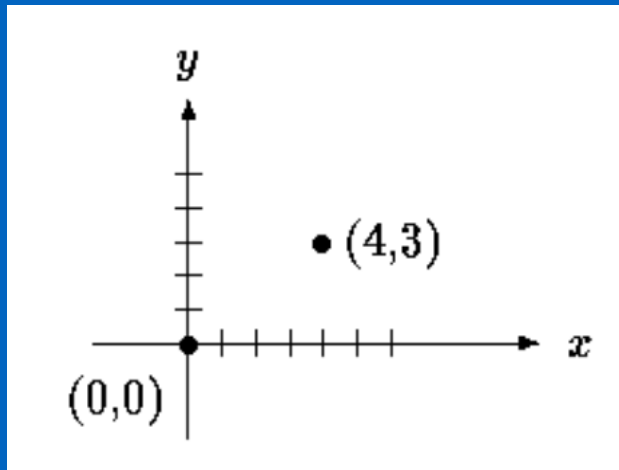
    scanf("%d%s%f",&student1.num,
          student1.name, &student1.score);
    scanf("%d%s%f",&student2.num,
          student2.name, &student2.score);

    printf("The higher score is:\n");
    if (student1.score>student2.score)
        printf("%d %s %6.2f\n",student1.num,
               student1.name, student1.score);
    else if (student1.score<student2.score)
        printf("%d %s %6.2f\n",student2.num,
               student2.name, student2.score);
    else{
        printf("%d %s %6.2f\n",student1.num,
               student1.name, student1.score);
        printf("%d %s %6.2f\n",student2.num,
               student2.name, student2.score);
    }

    return 0;
}
```


结构体变量的嵌套

- 结构体的成员是另一个结构体



- 一个矩形的坐标:

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

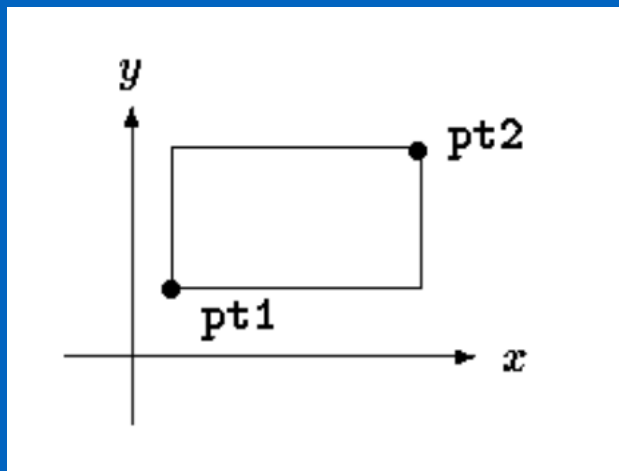
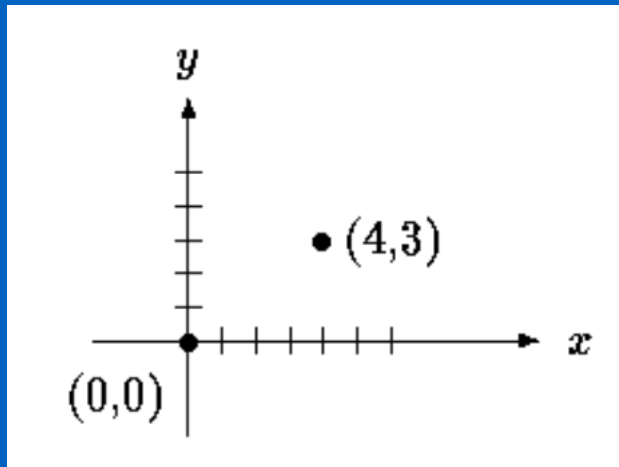
```
struct rect {  
    struct point pt1;  
    struct point pt2;  
};
```

```
struct rect screen;
```

```
//引用screen的成员pt1的x坐标  
screen.pt1.x = 4;  
screen.pt1.y = 3;  
...
```

结构体和函数

- 结构体作为函数的返回值



- 一个矩形的坐标:

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

struct rect {
    struct point pt1;
    struct point pt2;
};

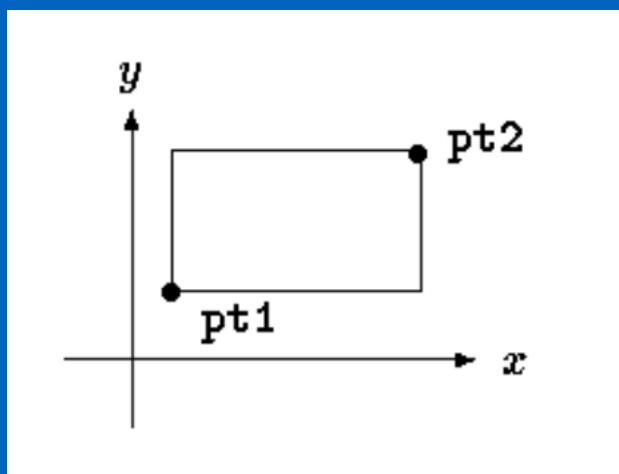
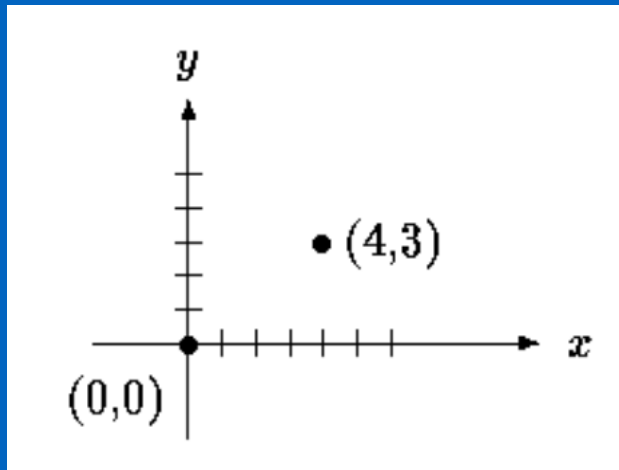
struct rect screen;

struct point makepoint(int x, int y)
{
    struct point temp;
    temp.x = x;
    temp.y = y;
    return temp;
}

screen.pt1 = makepoint(4,3);
screen.pt2 = makepoint(50,60);
```

结构体和函数

- 结构体作为函数的参数



- 两个点的坐标相加：

```
struct point {
    int x;
    int y;
} pt;

struct rect {
    struct point pt1;
    struct point pt2;
};

struct rect screen;

struct point makepoint(int x, int y)
{
    struct point temp;
    temp.x = x;
    temp.y = y;
    return temp;
}

screen.pt1 = makepoint(4,3);
screen.pt2 = makepoint(50,60);

struct addpoint(struct point p1, struct point p2) {
    p1.x += p2.x;
    p1.y += p2.y;
    return p1;
}

pt = addpoint(screen.pt1, screen.pt2);
```

结构体数组

- 有学生成绩表如下：

| | num | name | course_name | credit | grade | GP |
|------|------|------|-------------|--------|-------|-----|
| S[0] | 1001 | 李芳 | C 语言程序设计 | 4.0 | 85 | 3.5 |
| S[1] | 1001 | 李芳 | 高等数学（1） | 6.0 | 80 | 3.0 |
| S[2] | 1001 | 李芳 | 大学英语（1） | 4.0 | 75 | 2.5 |
| S[3] | 1002 | 赵力 | C 语言程序设计 | 4.0 | 90 | 4.0 |
| S[4] | 1002 | 赵力 | 高等数学（1） | 6.0 | 85 | 3.5 |
| S[5] | 1002 | 赵力 | 大学英语（1） | 4.0 | 80 | 3.0 |
| S[6] | 1003 | 王倩 | C 语言程序设计 | 4.0 | 60 | 1.0 |
| S[7] | 1003 | 王倩 | 高等数学（1） | 6.0 | 70 | 2.0 |
| S[8] | 1003 | 王倩 | 大学英语（1） | 4.0 | 80 | 3.0 |

- 每个数组元素都是一个结构类型的数据

```
//定义结构体数据类型
```

```
struct student {  
    int    num;  
    char   name[20];  
    char   course_name[20];  
    float  credit;  
    int    grade;  
    float  GP;  
}
```

```
//定义结构体数组
```

```
struct student s[9];
```

```
s[0].num = 1001;  
strcpy(s[0].name, "李芳");  
strcpy(s[0].course_name, "C语言程序设计");  
s[0].credit = 4.0;  
s[0].grade = 85;  
s[0].GP = 3.5;
```

```
//...
```

结构体数组

- 有学生成绩表如下：
- 建立两个结构体数组

| 学号 | 姓名 | 课程 | 学分 | 成绩 | 获得绩点 |
|------|----|---------|-----|----|------|
| 1001 | 李芳 | C语言程序设计 | 4.0 | 85 | 3.5 |
| 1001 | 李芳 | 高等数学（1） | 6.0 | 80 | 3.0 |
| 1001 | 李芳 | 大学英语（1） | 4.0 | 75 | 2.5 |
| 1002 | 赵力 | C语言程序设计 | 4.0 | 90 | 4.0 |
| 1002 | 赵力 | 高等数学（1） | 6.0 | 85 | 3.5 |
| 1002 | 赵力 | 大学英语（1） | 4.0 | 80 | 3.0 |
| 1003 | 王倩 | C语言程序设计 | 4.0 | 60 | 1.0 |
| 1003 | 王倩 | 高等数学（1） | 6.0 | 70 | 2.0 |
| 1003 | 王倩 | 大学英语（1） | 4.0 | 80 | 3.0 |

| | num | name | course_name | credit | grade | GP |
|------|------|------|-------------|--------|-------|-----|
| S[0] | 1001 | 李芳 | C 语言程序设计 | 4.0 | 85 | 3.5 |
| S[1] | 1001 | 李芳 | 高等数学（1） | 6.0 | 80 | 3.0 |
| S[2] | 1001 | 李芳 | 大学英语（1） | 4.0 | 75 | 2.5 |
| S[3] | 1002 | 赵力 | C 语言程序设计 | 4.0 | 90 | 4.0 |
| S[4] | 1002 | 赵力 | 高等数学（1） | 6.0 | 85 | 3.5 |
| S[5] | 1002 | 赵力 | 大学英语（1） | 4.0 | 80 | 3.0 |
| S[6] | 1003 | 王倩 | C 语言程序设计 | 4.0 | 60 | 1.0 |
| S[7] | 1003 | 王倩 | 高等数学（1） | 6.0 | 70 | 2.0 |
| S[8] | 1003 | 王倩 | 大学英语（1） | 4.0 | 80 | 3.0 |

- 计算每个学生平均绩点，得到如下平均绩点表：

| 学号 | 姓名 | GPA |
|------|----|-----|
| 1001 | 李芳 | 3.0 |
| 1002 | 赵力 | 3.5 |
| 1003 | 王倩 | 2.0 |

| | | | |
|----------|------|----|-----|
| S_GPA[0] | 1001 | 李芳 | 3.0 |
| S_GPA[1] | 1002 | 赵力 | 3.5 |
| S_GPA[2] | 1003 | 王倩 | 2.0 |

结构体数组

```
#include <stdio.h>

//定义结构体数据类型
struct student{
    int    num;
    char   name[20];
    char   course_name[20];
    float  credit;
    int    grade;
    float  GP;
} s[9];

struct student_GPA{
    int num;
    char name[20];
    float GPA;
} s_GPA[3];

int main(){
```

```
    int i,j;
    for(i=0;i<9;i++)
        scanf("%d%s%f%d%f",
            &s[i].num,s[i].name,
            s[i].course_name,&s[i].credit,
            &s[i].grade,&s[i].GP
        );

    //计算每个同学GPA
    for(i=0,j=0;i<9;i=i+3,j=j+1) {
        s_GPA[j].num=s[i].num;
        strcpy(s_GPA[j].name,s[i].name);
        s_GPA[j].GPA=(s[i].credit*s[i].GP
            +s[i+1].credit*s[i+1].GP
            +s[i+2].credit*s[i+2].GP)
            /(s[i].credit
            +s[i+1].credit
            +s[i+2].credit
        );
    }

    for(i=0;i<3;i++)
        printf("%d\t%s\t%.1f\n",
            s_GPA[i].num,
            s_GPA[i].name,
            s_GPA[i].GPA
        );

    return 0;
}
```

结构体数组

- 有学生成绩表如下：

| | num | name | course_name | credit | grade | GP |
|------|------|------|-------------|--------|-------|-----|
| S[0] | 1001 | 李芳 | C 语言程序设计 | 4.0 | 85 | 3.5 |
| S[1] | 1001 | 李芳 | 高等数学（1） | 6.0 | 80 | 3.0 |
| S[2] | 1001 | 李芳 | 大学英语（1） | 4.0 | 75 | 2.5 |
| S[3] | 1002 | 赵力 | C 语言程序设计 | 4.0 | 90 | 4.0 |
| S[4] | 1002 | 赵力 | 高等数学（1） | 6.0 | 85 | 3.5 |
| S[5] | 1002 | 赵力 | 大学英语（1） | 4.0 | 80 | 3.0 |
| S[6] | 1003 | 王倩 | C 语言程序设计 | 4.0 | 60 | 1.0 |
| S[7] | 1003 | 王倩 | 高等数学（1） | 6.0 | 70 | 2.0 |
| S[8] | 1003 | 王倩 | 大学英语（1） | 4.0 | 80 | 3.0 |

- 每个数组元素都是一个结构类型的数据

//定义结构体数据类型

```
struct student {  
    int    num;  
    char   name[20];  
    char   course_name[20];  
    float  credit;  
    int    grade;  
    float  GP;  
} s[9] = {  
    {1001, "李芳", "C语言程序设计", 4.0, 85, 3.5},  
    {1001, "李芳", "高等数学(1)", 6.0, 80, 3.0},  
    {1001, "李芳", "大学英语(1)", 4.0, 75, 2.5},  
    {1002, "赵力", "C语言程序设计", 4.0, 90, 4.0},  
    {1002, "赵力", "高等数学（1）", 6.0, 85, 3.5},  
    {1002, "赵力", "大学英语（1）", 4.0, 80, 3.0},  
    {1003, "王倩", "C语言程序设计", 4.0, 60, 1.0},  
    {1003, "王倩", "高等数学(1)", 6.0, 70, 2.0},  
    {1003, "王倩", "大学英语（1）", 4.0, 80, 3.0}  
};
```

输入三个人的姓名，性别和年龄，再分别输出。

```
#include<stdio.h>
#include<string.h>

struct person{
    char name[50];
    char sex;
    int age;
};

int main( ){
    struct person per[5] ;
    int k;

    for(k=0;k<=2;k++) {
        /*gets读取名字,直到回车结束(含空格)*/
        gets(per[k].name);
        scanf("%d%c",&per[k].age,&per[k].sex);
        /*getchar读取输入性别后的回车符*/
        getchar();
    }

    for(k=0;k<=2;k++){
        puts(per[k].name);
        printf("%c,%d\n",per[k].sex,per[k].age);
    }

    return 0;
}
```

运行时若输入：

HUANG

40F

WANG

30M

LIU

50F

则输出：

HUANG

F,40

WANG

M,30

LIU

F,50

- 有n个学生，每个学生的数据包括学号、姓名和3门课的成绩，用键盘输入n个学生的数据，要求打印出每个学生3门课的平均成绩，以及最高分的学生的数据（学号、姓名、3门课的成绩和平均分数）

```
#include<stdio.h>
#include<string.h>
#define MAX 100
struct student{//结构类型定义
    char name[20];
    int number;
    int s[3];
    int average;
};
struct student stu[MAX]; //定义结构数组

int main( ){

    int n,i,j,m,max;
    printf("请输入人数: ");
    scanf("%d",&n); //输入学生人数
    printf("请输入每个学生的姓名、学号以及三门课程成绩: \n");

    for (i=0;i<n;i++){ //输入学生数据
        printf("请输入第%d人数据\n",i+1);
        scanf("%s%d",stu[i].name,&stu[i].number);
        for(j=0;j<3;j++)
            scanf("%d",&stu[i].s[j]);
    }
```

```
        for(i=0;i<n;i++) { //输出学生数据
            printf("%s %d ", stu[i].name, stu[i].number);
            for(j=0;j<3;j++)
                printf("%d ",stu[i].s[j]);
            printf("\n");
        }

        for(i=0;i<n;i++) { //计算每个学生的平均分
            stu[i].average=0;
            for(j=0;j<3;j++)
                stu[i].average+=stu[i].s[j];
            stu[i].average= stu[i].average/3;
        }

        max=stu[0].average;
        m=0;
        for(i=1;i<n;i++) //查询平均分数最高的学生
            if (stu[i].average>max){
                max=stu[i].average;
                m=i;
            }

        printf("平均分最高的学生数据: \n");
        printf("%s %d %d %d %d %d ", stu[m].name,
            stu[m].number, stu[m].s[1], stu[m].s[2],
            stu[m].s[3], stu[m].average);

        return 0;
    }
```