# 任务07 数组和广义表

**任务07\_1 数组的顺序表示和实现**

编写函数ContainSameElem判断一个m行n列的二维数组中是否存在相同的元素。

**函数原型：**

int ContainSameElem(int \*a,int m,int n) ：若二维数组中有相同元素返回1，并输出相同元素值，及下标位置，详见运行效果；如果没有相同元素返回0。

**编写要求：**

1. 请用C语言完成函数编写，且符合C89语法规则（影响成绩评定）

2. 运行效果必须严格符合要求（影响成绩评定）

3. 需要提交ContainSameElem函数

**main函数如下：**

int main(void)

{

int a[4][3]={11, 12, 13, 24, 25, 26, 13, 38, 39, 40, 39, 42};

if(ContainSameElem(a[0],4,3))

printf("a有相同元素!\n");

else

printf("a无相同元素\n");

return 0;

}

**运行效果：**

a[0][2]：13

a[2][0]：13

a[2][2]：39

a[3][1]：39

a有相同元素!

请按任意键继续. . .

**提交格式：**

ContainSameElem函数如下：

# 任务07\_2 三元组顺序表

请根据要求，按照**C语言语法，**编写三元组顺序表的两个函数：JudgingSymmetricMatrix、PrintMatrix，完成规定的运行效果。

**预定义常量和类型**

#define Status int

#define OK 1

#define ERROR 0

#define MAXSIZE 12500

**三元组顺序表结构**

typedef struct{

int i,j; //行数，列数

int e; //非零元素值

}Triple;

typedef struct{

Triple data[MAXSIZE + 1];

int mu,nu,tu;

}TSMatrix;

**函数原型及功能如下：**

Status JudgingSymmetricMatrix(TSMatrix M) ：判断矩阵M是否是对称矩阵，如果是，返回OK，如果不是，返回ERROR;

void PrintMatrix(TSMatrix M)：将矩阵M打印输出，输出效果见运行效果。

**编写要求：**

1. 请用C语言完成函数编写，且符合C89语法规则（影响成绩评定）

2. 运行效果必须严格符合要求（影响成绩评定）

3. 提交JudgingSymmetricMatrix、PrintMatrix这两个函数

**（1）若main函数如下：**

int main(void)

{

TSMatrix M;

M.mu = 4; M.nu = 4; M.tu = 6;

M.data[1].i = 1; M.data[1].j = 1; M.data[1].e = 3;

M.data[2].i = 1; M.data[2].j = 4; M.data[2].e = 5;

M.data[3].i = 2; M.data[3].j = 2; M.data[3].e = -1;

M.data[4].i = 3; M.data[4].j = 4; M.data[4].e = 9;

M.data[5].i = 4; M.data[5].j = 1; M.data[5].e = 5;

M.data[6].i = 4; M.data[6].j = 3; M.data[6].e = 9;

if(JudgingSymmetricMatrix(M))

printf("是对称矩阵！\n");

else

printf("不是对称矩阵！\n");

PrintMatrix(M);

return 0;

}

**运行效果：**

是对称矩阵！

3 0 0 5

0 -1 0 0

0 0 0 9

5 0 9 0

请按任意键继续. . .

**（2）若main函数如下：**

int main(void)

{

TSMatrix M;

M.mu = 4; M.nu = 4; M.tu = 4;

M.data[1].i = 1; M.data[1].j = 1; M.data[1].e = 3;

M.data[2].i = 1; M.data[2].j = 4; M.data[2].e = 5;

M.data[3].i = 2; M.data[3].j = 2; M.data[3].e = -1;

M.data[4].i = 3; M.data[4].j = 1; M.data[4].e = 2;

if(JudgingSymmetricMatrix(M))

printf("是对称矩阵！\n");

else

printf("不是对称矩阵！\n");

PrintMatrix(M);

return 0;

}

**运行效果：**

不是对称矩阵！

3 0 0 5

0 -1 0 0

2 0 0 0

0 0 0 0

请按任意键继续. . .

**提交格式：**

JudgingSymmetricMatrix函数如下：

PrintMatrix函数如下：

# 任务07\_3 画图题

（1）请画出矩阵M的十字链表存储结构图：

（2）请分别用**首尾链表法**和**扩展线性链表（孩子兄弟链表）**画出广义表A的存储结构：

A=(a, ((b, c, d)),(e))

**提交方式：**

1. 矩阵M的十字链表存储方式结构图如下：
2. 广义表A的**首尾链表法**存储图如下：

广义表A的**扩展线性链表（孩子兄弟链表）**存储图如下：