

北京交通大学

《数据结构（A）》第 5 章作业

设计型作业

专 业： 计算机科学与技术

班 级：

学生姓名：

学 号：

北京交通大学计算机与信息技术学院

2021 年 11 月 11 日

《数据结构 (A)》第 5 章设计型作业^①

1 设计型题目

5.1 鞍点问题：若矩阵 A 中的某一元素 $A[i,j]$ 是第 i 行中的最小值，而又是第 j 列中的最大值，则称 $A[i,j]$ 是矩阵 A 中的一个鞍点。写出一个可以确定鞍点位置的程序。

5.2 稀疏矩阵转置：输入稀疏矩阵中每个元素的行号、列号、值，建立稀疏矩阵的三元组存储结构，并将此矩阵转置，显示转置前后的三元组结构。

5.3 （选作）用头尾链表存储表示法建立广义表，输出广义表，求广义表的表头、广义表的表尾和广义表的深度。

2 设计型题目解答

【5.1 题解答】：

思路：

正常建立二维矩阵，再建立两个数组保留每行最小值，每列最大值。两层循环，找到每行最小值，每列最大值。最后对比两个数据的数，相等则输出其坐标和值。

代码：

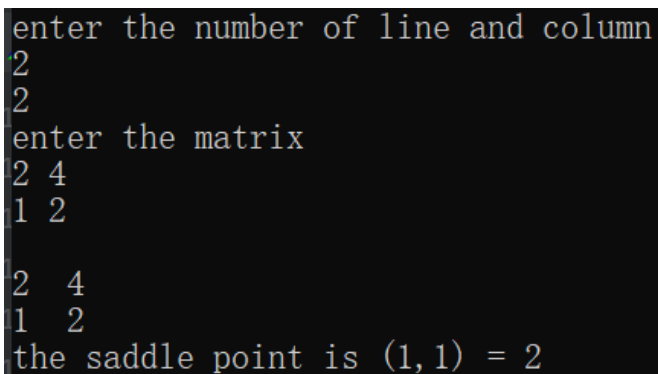
```
01:
02:#include <stdio.h>
03:#define M 10
```

^① 这是《数据结构 (A)》第 5 章的设计型作业，学生提交的截止日期是 2021 年 11 月 14 日。

```
04:#define N 10
05:
06:int main() {
07:
08:    int matrix[M][N]={0};
09:    int m, n;
10:    int min[M]={0};
11:    int max[N]={0};
12:    int flag;
13:    printf("enter the number of line and column\n");
14:    scanf("%d%d", &m, &n);
15:
16:
17:    printf("enter the matrix\n");
18://建立矩阵
19:    for(int i=0; i<m; ++i)
20:        for(int j=0; j<n; ++j)
21:            scanf("%d", &matrix[i][j]);
22:
23:    printf("\n");
24:    for(int i=0; i<m; ++i)
25:    {
26:        for(int j=0; j<n; ++j)
27:        {
28:            printf("%d ", matrix[i][j]);
29:        }
30:        printf("\n");
31:    }
32://找到每行最小值
33:    for(int i=0; i<m; ++i)
34:    {
35:        min[i]= matrix[i][0];
36:        for(int j=0; j<n; ++j)
37:        {
38:            if(matrix[i][j]<min[i])
39:            {
40:                min[i]= matrix[i][j];
41:            }
42:        }
43:    }
44://找到每列最大值
45:    for(int j=0; j<n; ++j)
46:    {
```

```
47:     max[j]= matrix[0][j];
48:     for(int i=0; i<m; ++i)
49:     {
50:         if(matrix[i][j]>max[j])
51:             max[j] = matrix[i][j];
52:     }
53:
54:
55: }
56://对比找到重合部分
57: for(int i=0; i<m; ++i)
58: {
59:     for(int j=0; j<n; ++j)
60:     {
61:         if(max[j]==min[i])
62:         {
63:             flag =1;
64:             printf("the saddle point is (%d,%d) = %d",i+1,j+1, min[i]);
65:         }
66:     }
67:
68:
69: }
70: if(flag==0)
71:     printf("no saddle point\n");
72:
73: return 0;
74:
75:}
```

调试结果:



```
enter the number of line and column
2
2
enter the matrix
2 4
1 2

2 4
1 2
the saddle point is (1,1) = 2
```

图 5.1

调试正常

【5.2 题解答】:

思路:

以三元组建立矩阵，从控制台输入矩阵行列数，和非零元素数。再建立三元组，从控制台键入三元组参数。接着交换三元组中元素的行数和列数，每个元素的行列参数，即为转置。

代码:

```
01:#include <stdio.h>
02:#define MAX 100
03://三元组元素
04:typedef struct{
05:    int i,j;
06:    int e;
07:}triple;
08://三元组
09:typedef struct{
10:    int m, n, num;
11:    triple data[MAX];
12:}TSMatrix;
13://建立三元组矩阵 由控制台输入
14:TSMatrix &create_matrix(int m, int n, int t){
15:    TSMatrix M;
16:    M.m = m;
17:    M.n = n;
18:    M.num = t;
19:    printf("please enter the i, j, data of the matrix element\n");
20:    for(int i=0; i<t; ++i){
21:        scanf("%d%d%d", &M.data[i].i, &M.data[i].j, &M.data[i].e);
22:        while(M.data[i].i>m || M.data[i].j>n){
23:            printf("the parameter is too big! enter again\n");
24:            scanf("%d%d%d", &M.data[i].i, &M.data[i].j, &M.data[i].e);
25:        }
```

```
26:    }
27:    return M;
28:}
29://打印三元组
30:void show_matrix_triple(TSMatrix &M){
31:    for(int i= 0; i<M.num; ++i){
32:        printf("%3d %3d %3d\n",M.data[i].i, M.data[i].j, M.data[i].e);
33:    }
34:
35:}
36://转置
37:TSMatrix & transpose_matrix(TSMatrix &M){
38:    int tem;
39:    for(int k=0; k<M.num;++k) {
40:        tem = M.data[k].i;
41:        M.data[k].i = M.data[k].j;
42:        M.data[k].j = tem;
43:    }
44:    return M;
45:}
46:
47:int main()
48:{
49:    int m,n , num;
50:    printf("please enter the parameter\n");
51:    scanf("%d %d %d", &m, &n, &num);
52:    TSMatrix M= create_matrix(m, n , num);
53:    show_matrix_triple(M);
54:    printf("\n");//转置前后 空一行 方便查看
55:    transpose_matrix(M);
56:    show_matrix_triple(M);
57:}
```

调试:

调试结果正常

```
please enter the parameter
2 3
3
please enter the i, j, data of the matrix element
1 3 4
2 2 5
3 1 4
the parameter is too big! enter again
2 3 6
1 3 4
2 2 5
2 3 6
3 1 4
2 2 5
3 2 6
```

图 5.2