

第五章作业答案

5.8

当 i 为奇数时, $k=i+j-2$

当 i 为偶数时, $k=i+j-1$

$k=i+j-i\%2-1$

//这题答案形式比较多, 也可以从 4 个元素为一个块来归纳。

//但注意 k 是从 0 开始的。

5.26

```
typedef struct OLNode{
    int i,j;
    ElemType e;
    struct OLNode *right,*down; //元素所在行表和列表的后继链域
}OLNode,*OLink;
```

```
typedef struct{
    OLink *rhead,*thead; //行和列链表的头指针
    int mu,nu,tu; //稀疏矩阵的行数, 列数和非零元素个数
}CrossList;
```

```
Status PrintCrossList(CrossList M){
    //以三元组形式输出用十字链表表示的稀疏矩阵中非零元素及其下标
    OLink p;
    if(M.tu==0)
        return ERROR;
    for(int i=0;i<M.mu;i++){
        p=M.rhead[i];
        while(p!=NULL){
            printf("%d %d %d\n",p->i,p->j,p->e);
            p=p->right;
        }
    }
    return OK;
}
```

//有的同学的写法, 是把十字链表的非零元素以三元组形式存起来。

//题目要求是输出。

//广义表定义如下

```
typedef enum { Head, Tail }Mark;
typedef enum { Atom, List }ElemTag;
typedef char AtomType;
typedef struct GLNode
{
```

```

union { int mark; };
ElemTag tag;
union
{
    AtomType atom;
    struct
    {
        struct GLNode *hp;
        struct GLNode *tp;
    }ptr;
}Union;
}GLNode;
typedef GLNode* GList;

int problem_5_30(GList L)
{
    int m, n;
    if (!L)return 1;//注意!!!空表深度为1
    if (L->tag == Atom)return 0;
    m = problem_5_30(L->Union.ptr.hp) + 1;
    n = problem_5_30(L->Union.ptr.tp);
    return m > n ? m : n;
}

bool problem_5_32(GList A,GList B)
{
    if (!A && !B)
        return true;

    if (A && B)
//注意!!!因为是递归写法，所以一定要判断是否为空否则程序没有终止会进入死循环
    {
        if (A->tag == B->tag)
        {
            if (A->tag == Atom)
                if(A->Union.atom==B->Union.atom)
                    return true;
            else
                if(problem_5_32(A->Union.ptr.hp, B->Union.ptr.hp) &&
problem_5_32(A->Union.ptr.tp, B->Union.ptr.tp))
                    return true;
        }
    }
}

```

```
        return false;  
    }
```