5.8

5.7 解: (1) $k=2(i-1)+j-1=2i+j-3$

(2) $i=(k+1)/3+1,\ (0\le k\le 3n-1)$

$j=k+3-2i=k+1-2[(k+1)/3]$

5.8

解:

$i-j=0$ 时.

若 i 为奇数, j 为奇数, 则 $k=2(i-1)+1-1=2i-1-1$

若 i 为偶数, j 为偶数, 则 $k=2(i-1)+2-1=2i-0-1$

$i-j=1$ 时.

若 i 为偶数, j 为奇数. 则 $k=2(i-1)+1-1=2i-1-1$

$i-j=-1$ 时

若 i 为奇数, j 为偶数, 则 $k=2(i-1)+2-1=2i-0-1$

综上, $k=2i-j\%2-1$

5.10

解: (1) ~~原表~~ 结果 = P

(2) 结果 = (k, P, h)

(3) 结果 = (a, b)

(4) 结果 = ((c, d))

(5) 结果 = GetHead [((c, d))] = (c, d)

(6) 结果 = GetTail [(a, b)] = (b)

(7) 结果 = GetHead [GetTail [(a, b)]] = GetHead [(b)] = b

(8) 结果 = GetTail [GetHead [((c, d))]] = GetTail [(c, d)] = (d)

5.17

解：~~输出最大值~~

~~int Algo_1(SqList L, int len)~~

~~{~~

~~int max, tmp;~~

~~max = L.elem~~

```
//求最大值
int Max(SqList &L, int k)
{
  if(k<L.Length-1)
      if( L.elem[k]<Max(L,k+1))
          return Max(L, k+1);

      else
          return L.elem[k];

  else
      return L.elem[k];

}
//求最小值
 int Min (SqList &L, int k)
{
  if (k<L.Length-1)
      if(L.elem[k]>Min(L, k+1))
          return Min(L, k+1);

      else
          return L.elem[k];

  else
      return L.elem[k];
}
```

第四次作业

王小龙

20202115O2

3/7

```
//求和
int Sum(SqList &L, int k)
{
    if(k==0)
        return L.elem[0];
    else
        return L.elem[k]+Sum(a,k-1);
}
//求积
int Product(SqList &L,int k)
{
    if(k==0)
        return L.elem[0];
    else
        return L.elem[k]*Sum(a,k-1);
}
//求平均值
double Avg(SqList &L, int k)
{
    if(k==0)
        return L.elem[0];
    else
        return (Avg(a,k-1)*k+L.elem[k])/(k+1);
}
```

第四次作业

王小龙

20202115 02

4/7

5.18

解：

```
void Rotate (ArrayID &a, int n, int k)
{
    ElemType *p=a, temp;
    int i,j;
    if (0!=k%n){
        for(i=1; i<=k; ++i){
            temp=a[n-1];
            for (j=n-2; j>=0; --j){
                a[j+1]=a[j];
            }
            a[0]=temp;
        }
    }
}
```

5.21

解：

```
/*函数原型*/
Status Algo.(TSMatrix A, TSMatrix B, TSMatrix *C);
int main(int argc, char *argv[])
{
    TSMatrix A,B,C;
    FILE *fp;
    printf(
    fp=fopen("Data/Algo.txt", "r");
    CreatSMatrix_T(fp, 2, &A, &B);
    fclose(fp);
    printf("A=\n");
    PrintSMatrix_T(A);
    printf("B=\n");
    PrintSMatrix_T(B);
    printf("\n");
```

```
    Algo(A, B, &C);
    printf("C = A+B = \n");
    PrintSMatrix_T(C);
    printf("\n");

    return 0;

}
```

//三元矩阵加法

```
Status Algo(TSMatrix A, TSMatrix B, TSMatrix *C)
{
    if(AddSMatri_T(A,B,C))
        return OK;

    else
        return ERROR;
}
```

5.26.

```
void OutCSM(CrossList M, void(*Out3)(int,int,int))
//用函数Out3,依次以三元组格式输出十字链表表示的矩阵
{
    int i=1, row, col, e;
    OLink p,q;
    for(i=1; i<=M.mu; ++i){
        p=M.rhead[i];
        while(p){
            Out3(p->i, p->j, p->e);
            p = p->right;
        }
    }
}
```

第四次作业
王小龙
20202111502
6/7

5.34
解：

```
/*函数原型*/
void Algo(GList *L);
int main(int argc, char*argv[])
{
  GList Li
  char *s= "(a,((b,c),()),(((d),e),f))";
  SString S;
  StrAssign_Sq(S,s);
  CreateGList_GL-H-T-1(&L,S);
  printf("L=");
  Out put_GL-H-T(L.Head);
  printf("\n\n");
  Algo(&L);
  printf("L=");
  Output_GL_H_T(L,Head);
  printf("\n\n");
  return 0;
}
void Algo(GList *L)
{
  GList head,tail;
  if(*L)
  {
    head=*L;
    tail=(*L)->Union.ptr.tp;
    if(head->Union.ptr.hp && head->Union.ptr.hp->tag==List)
      Algo(&(head->Union.ptr.hp));
    if(tail)
    {
```

~~Algo2~~

```
Algo(&((*L)->Union.ptr.tp));
    *L = (*L)->Union.ptr.tp;
    tail->Union.ptr.tp;
    tail->Union.ptr.tp = head;
    head->Union.ptr.tp = NULL;
  }
 }
}
```

王小龙

学号

第四次作业

20202111502

7/7