5.9 、稀疏矩阵 n×n. 有m个非零元

时间上：三元组 O(m). 二维数组 O(n^2)

空间上 三元组存储量 远远小于 二维数组

$m << n \times n$

5.17 ③ (1)

```
Status   Max(List &L, Status i)
{ if(i< L.length-1
   {if (L.elemt[i] <Max(L,i+1)
          return Max(L,i+1);
    else return L.elemt[i];
   }
}

Status  Min(List &L, Status i)
{ if( i<L.length-1)
   { if(L.elemt[i] > Min(L,i+1)
       return min(L,i+1);
     else
       return (L.elemt[i]);
   }
}

Status  Sum(List &L, Status i)
{ if( i==0)
       return L.elemt[0];
  else return L.elemt[i] + Sum(L, i-1)
}
```

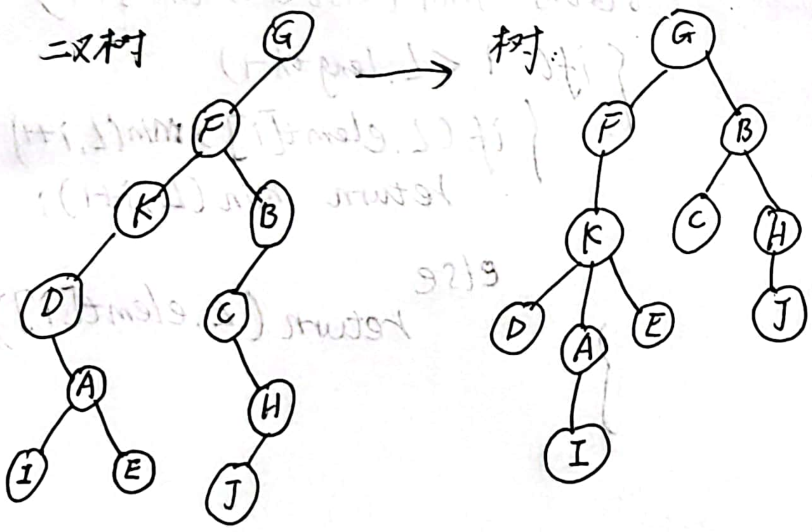
6.14

(1) 不含左子树的树

(2) 不含右子树的树

(3) 只有根结点的树

6.23



6.36.

```
Status  STree(BiTree &T1, BiTree &T2)
{ if(!T1)
  { if(!T2) return ERROR;
  }
  else
  { if(!T2)
        return ERROR;
    else
    { if( STree(T1->lchild, T2->lchild)
          && STree(T1->rchild, T2->rchi
        return OK;
      else return ERROR;
```

6.37.

```
Status  POT( BiTree &T. Status(ElemType e)))
  {
    Stack  S
    Init Stack(s)

    while ( T || ! StackEmpty(s))
        { if( T )
          { if(! Visit(T→data))
                return ERROR;
            Push( s, T→ rchild)
            T =T →lchild; }
          else  Pop(s, T )
        } return OK;
    }
```

```
6.43.   Status  EBiTree(BiTree &T)
 {  BiTree  T2 ;
    if(T)
      { T2 = T→lchild;
        T→lchild =T→rchild;
        T→rchild  = T2 ;
          EBiTree(T→lchild);
        EBiTree(T→rchild);
      }
      return  OK;
 }
```