1.
```
void Del_X(LinkList &L, ElemType x)
{ if(L==NULL) return;
  if(L->data==x)
  {LNode *p=L;
   L=L->next;
   free(p);
   Del_X(L,x);
  }
  else Del_X(L->next,x);
}
```

2.
```
void del_x(LinkList &L, ElemType x)
{ LinkList *p=L->next;
  LinkList *r=L;
  while(p!=NULL)
  { if(p->date==x)
    { LinkList *q=p;
      r->next=p->next;
      p=p->next;
      free(q);
    }
    else
    {p=p->next;
     r=r->next;
    }
  }
}
```

```
3. void reverse(LinkList &L)
{ LinkList *L2; node *p1=L,*s;
  while(p1->next!=NULL)
  { s=(LinkList)malloc(sizeof(node));
    s->data=p1->next->data;
    s->next=L2->next;
    L2->next=s;
    p1=p1->next;
  }
}
4. void Del_min(Link &L)
{ LinkList *p=L->next;
  LinkList *r=L;
  LinkList *q=p;
  while(p->next!=NULL)
  { if(p->next->data<q->data)
    { r=p; q=p->next;
    }p=p->next;
  }
  r->next=q->next;
  free(q);
}
5. void reverse(LinkList &L)
{ LinkList *p=L->next->next;
  LinkList *q;
  L->next->next=NULL;
  while(p)
  { q=p->next;
    p->next=L->next;
    L->next=p;
    p=q;
  }
}
```

6、
```
void sort(LinkList &L)
{ LinkList *p=L->next, *pre;
  LinkList *r=p->next;
  p->next=NULL;
  p=r;
  while(p!=NULL)
  { r=p->next;
    pre=L;
    while(pre->next!=NULL&&pre->next->data<p->data)
    pre=pre->next;
    p->next=pre->next;
    pre->next=p;
    p=r;
  }
}
```

7、
```
void Del_x1_x2(LinkList &L, ElemType x1, ElemType x2)
{ LinkList *p1=L->next;
  LinkList *p2=L->next;
  while(p1->next!=NULL)
  { p1=p1->next; p2=p1;
    if(p1->next->data==x1)
    { while(p2!=NULL)
      { p2=p2->next;
        if(p2->data==x2)
        p1->next=p2->next;
      }
    }
  }
}
```

8.
```
void find(LinkList &L1, LinkList &L2)
{LinkList *p1=L1->next;
 LinkList *p2=L2->next;
 while(p1!=NULL)
 {p1=p1->next; p2=L2->next;
  while(p2!=NULL)
  {p2=p2->next; p2=L2->next;
   if(p1->next=p2->next)
   {cout<<p1->next->data;
    break;
   }
   p2=p2->next;
  }
  p1=p1->next;
 }
}
```

9.
```
void sortdel(LinkList *&L)
{LinkList *p,*pre,*minp,*minpre;
 LinkList *temp;
 while(L->next!=NULL)
 {p=L->next;
  pre=L;
  minpre=pre;
  minp=p;
  while(p!=NULL)
  {if(p->data<minp->data)
   {minpre=pre;
    minp=p;
   }
  }
  p=p->next;
  pre=pre->next;
 }
```

```
        temp=minp;
        minpre->next=minp->next;
        cout<<temp->data<<" ";
        free(temp);
    }
    free(L);
}
```

ADD:FUJIAN XIAMEN CABLE:0633 P.C:361005

~~总习题~~
~~2.设~~

```
10. void resolve(LinkList &LA, LinkList &LB)
 { LB = new node;
   LinkList *pa = LA -> next, *ra = pa;
   LinkList *pb = LB -> next, *rb = pb;
   while (pa)
   { if (pa -> data % 2 != 0)
         pa = pa -> next;  { ra = pa;
                             pa = pa -> next;
                           }
     else
       { pb = new node;
         pb -> data = pa -> data;
         rb -> next = pb;
         pb -> next = NULL;
         rb = pb;
         pa = pa -> next;
         ra -> next = ra -> next -> next;
         pa = ra -> next;
       }
   }
 }
```