3.4

解：(1) 栈中的数据元素逆置

(2) 如果栈中存在元素e，将其从栈中清楚

3.13

解：队 功能为队列逆置

3.14

解：(1) 4|32

(2) 42|3

(3) 423|

3.15

解：~~初始化双向栈~~

```
void Inistack_TWS_3_15(TWStack *S)
{
   (*S).t[Left]=-1;
   (*S).t[Right]=N;
}
Status Push_TWS_3_15(TWStack *S, StackName name, SElemType x)
{
   if((*S).t[Left]+1==(*S).t[Right])
      return ~~error~~ERROR;
   switch(name)
   {case Left:
        (*s).t[name]++;
        break;
    case Right:
        (*s).t[name]--;
        break;
   }
```

```
    }
    (*S).s[(*S).t[name]]=x;

    return  OK;
}
Status Pop_TWS_3_15(TWStack*S, StackName  name, SElemType*x)
{
 switch(name)
  {
    case Left:
        if ((*S).t[name]==-1)
           return ERROR;
        *x=(*S).s[(*S).t[name]];
         (*S).t[name]--;
        break;
case Right:
        if((*S).t[name]==N)
          return ERROR;
        *x=(*S).s[(*S).t[~~a~~ name]];
        (*S).t[name]++;
        break;
  }
}
```

3.28

王小龙
20202111502
第二次作业
3/5

解：
```
Status InitQueue(LinkQueue *Q)        //队列初始化
{
  (*Q).rear = (QueuePtr)malloc(sizeof(QNode));
  if(!(*Q).rear)
     exit(OVERFLOW);

  (*Q).rear -> next = (*Q).rear;

  return OK;
}
Status EnQueue(LinkQueue *Q, QElemType e)     //入队
{
  QueuePtr p;
  p = (QueuePtr)malloc(sizeof(QNode));
  if(!p)
     exit(OVERFLOW);
  p -> data = e;
  p -> next = (*Q).rear -> next;
  (*Q).rear -> next = p;
  (*Q).rear = p;
  return OK;
}
Status DeQueue(LinkQueue *Q, QElemType *e)    //出队
{
  QueuePtr h, p;
  h = (*Q).rear -> next;
  if (h -> next == (*Q).rear -> next)
     return ERROR;
  p = h -> next;
  *e = p -> data;
  h -> next = p -> next;
  if (p == (*Q).rear)
     (*Q).rear = h;
  free(p);
  return OK;
}
```

王小龙
20202115O2
第二次作业
4/5

3.29
解：~~Status EnQueue(SqQueue *Q, QElemType e)~~
Status EnQueue(CTagQueue &Q, QElemType x)

```
{
  if(Q.tag){
      return ERROR;
  }
Q.elem[Q.rear]=x;
if(Q.rear==MAXQSIZE-1){
   Q.rear=0;
}
else{
    ++Q.rear;
}
if(Q.rear==Q.front){
   Q.tag=1;
 }
 return OK;
}
Status DeCQueue(CTagQueue &Q, QElemType &x)
{  if(Q.front==Q.rear&&0==Q.tag){
    return ERROR;
     }
    else{
       x=Q.elem[Q.front];
      if(Q.front!=MAXQSIZE-1){
        ++Q.front;
      }
      else{
        Q.front=0;
      }if(Q.front==Q.rear){
          Q.tag=0;
```

王小龙
20202111502
第二次作业
5/5

```
    }
    return OK;
}
```