

## 4. 双向链表操作实现

给定一个序列，实现双向链表数据结构的添加、查找、删除、统计、翻转操作 输入由两部分组成，原始创建链表数据和依次执行的操作 执行的操作包括addbegin、addend、addmid、search、delete、count、reverse,每个操作打印对应输出 addbegin val，在序列开头插入val，打印新链表

addend val，在序列末尾插入val，打印新链表

addmid val pos，在序列pos位置插入val，打印新链表，pos从0开始计数，不会有超出链表长度的pos

search val，查找val的位置，打印位置pos，如果有链表中有多个相同值，打印第一个值，如链表为111，search 1，则输出0

delete pos，删除pos位置的节点，打印新链表，不会有超出链表长度的pos

count，打印链表元素个数

reverse，翻转整个链表

双链表结构

```
struct node
{
    int val;
    struct node *next;
    struct node *prev;
}
```

示例

输入：

1 2 3 4 5 6

addbegin 0

addend 7

addmid 10 3

search 4

delete 4

count

reverse

输出：

0 1 2 3 4 5 6

0 1 2 3 4 5 6 7

0 1 2 10 3 4 5 6 7

5

0 1 2 10 4 5 6 7

8

7 6 5 4 10 2 1 0

