

백준2346 - 풍선 터뜨리기

```
public class Baekjoon_2346 {  
    public static void main(String[] args) throws IOException {  
        BufferedReader br = new BufferedReader(new InputStreamReader(System.in)); // 버퍼 선언, 생성  
        int N = Integer.parseInt(br.readLine()); // 풍선의 수 입력받기  
        Deque<pair> deque = new LinkedList<>(); //pair 타입 Deque 선언  
        StringBuilder sb = new StringBuilder(); // 스트링빌더 생성  
  
        StringTokenizer st = new StringTokenizer(br.readLine()); // 풍선의 정보를 받기 위해서 st 생성  
  
        for (int i = 1; i <= N; i++) {  
            deque.add(new pair(i, Integer.parseInt(st.nextToken()))); // 풍선 정보를 deque에 추가할 때 몇번 째 풍선인지와 함께 풍선안의 내용도 같이 추가  
        }  
  
        while (!deque.isEmpty()) {  
            sb.append(deque.getFirst().index+" "); // 첫번째 풍선 인덱스 저장  
            int next = deque.poll().next; // 다음 풍선을 알기 위해서 풍선 내부에 저장되어있는 내용 확인  
  
            if (deque.isEmpty()) // 모든 풍선이 터졌으면 반복문 중지  
                break;  
            |  
            if (next > 0) { // 앞에 내용물이 양수  
                next -= 1; // 풍선이 터지면서 한칸 씩 당겨지므로 next -1 해주기  
                for (int i = 0; i < next ; i++) { // 남은 텍스트만큼 왼쪽으로 이동하기  
                    deque.addLast(deque.pollFirst());  
                }  
            }  
            if (next <= 0) { // 앞에 내용물이 음수  
                for (int i = 0; i < Math.abs(next) ; i++) { // 남은 텍스트만큼 오른쪽으로 이동하기  
                    deque.addFirst(deque.pollLast());  
                }  
            }  
        }  
        System.out.println(sb.toString()); // 최종출력  
    }  
}  
  
class pair { // 배열안에 2개의 정보를 넣기 위한 pair 클래스  
    int index; // 몇 번째 풍선인지  
    int next; // 풍선 안에 내용이 무엇인지  
  
    pair(int index, int next) { // pair 생성자  
        this.index = index;  
        this.next = next;  
    }  
}
```