

Решите задачу:

Напишите функцию `createList()`, на вход которой подается отсортированный по возрастанию слайс из целых чисел. Эта функция внутри должна создать цепочку структур. По сути это является связным списком.

```
type ListNode struct {  
    Val int  
    Next *ListNode  
}
```

Где `Val` будет принимать значения из переданного в функцию слайса чисел. А в `Next` соответственно должна попасть следующая созданная структура.

В результате функция должна вернуть структуру `ListNode` из начала цепочки.

Вторая функция - `deleteDuplicates()`. На вход она принимает начало вашего связного цикла и удаляет из цепочки структуры с одинаковыми `Val`.

Например, если мы вызовем структуру вот так

```
s := createList([]int{1,2,3,3})
```

то в `s` должна вернуться структура `ListNode` (начала цепочки), у нее значение `Val` = 1, а в `Next` - структура `ListNode`, у которой значение 2 и т.д все 4 структуры.

И после вызова `deleteDuplicates(s)` в который мы передадим наше начало цепочки, удаляется последний элемент со значением 3, так как это значение дублируется.

И в результате вернется структура `ListNode` - начало все той же цепочки, но в ней уже на одно значение меньше.