

1. 反转链表

```
let data = [1, 2, 3, 4, 5, 6, 7, 8, 9]
function reversaldata(k, data) {
    let Knumber = k - 1
    for (let i = 0; i < data.length; i++) {
        if (i >= Knumber) {
            i = i + Math.ceil(k / 2)
            Knumber = i + k - 1
        }
        if (Knumber >= data.length) {
            return data
        }
        if (i < Knumber) {
            let number = data[i]
            data[i] = data[Knumber]
            data[Knumber] = number
            Knumber--
        }
    }
    return data
}
console.log(reversaldata(4, data))
```

2. 两数相加：关键点：最后一次相加的进位

```
var addTwoNumbers = function (l1, l2) {
    let head = null
    let body = null
    let carry = 0
    while(l1 || l2){
        let n1 = l1 ? l1.val : 0
        let n2 = l2 ? l2.val : 0
        let numbers = n1 + n2 + carry
        if(!head){
            head = body = new ListNode(numbers % 10)
        }else{
            body.next = new ListNode(numbers % 10)
            body = body.next
        }
        carry = Math.floor(numbers/10)
        if(carry > 0){
            body.next = new ListNode(carry)
        }
        l1 = l1 && l1.next
        l2 = l2 && l2.next
    }
    return head
};
```

3. 无重复字符的最长子串：关键点：每次循环将数组或 set 从开头删除一个，确保有新的子串

```
var lengthOfLongestSubstring = function (s) {  
  let mylength = s.length  
  let arr = []  
  let j = 0  
  let maxlength = 0  
  for (let i = 0; i < mylength; i++) {  
    if (i !== 0) {  
      arr.shift()  
    }  
    while (j < mylength && !arr.includes(s[j])) {  
      arr.push(s[j])  
      j++  
    }  
    maxlength = Math.max(maxlength, arr.length)  
  }  
  return maxlength  
};
```