1 . 给定一个有序数组、从中间截断如1，2，3，4，5，6，7 。截断转置 5671234， 实现一个查找算法。

target = 4 ,返回4，在截断之后数组的位置

input ： [5，6，7，1，2，3，4] ，target = 4 output：6

function findIndex (arr，target) {

let l = 0,r = arr.length -1;

while(l <= r){

let mid = ((r - l ) >> 1) + l;

if(arr[mid] < target && target >= arr[l]){

r = mid -1;

}else if(arr[mid] <= target && target <= arr[r]){

l = mid;

}

}

return l;

}

1. 单链表：删除倒数第K个节点

// 快慢指针

function Node (element) {

this.value = element;

this.next = null;

}

fuction del (nodeList,k) {

let head = new Node('head');

head.next = nodeList;

let l = r = head ;

for(let i = 0 ; i < k+1 ;i++){

r = r.next;

}

while(r){

l = l.next;

r = r.next;

}

l.next = l.next.next;

return head.next;

}

3 . 简述一下js的原型与原型链

function A () {

}

const obj = new A();

obj.\_\_proto\_\_ === A.prototype

 A.prototype.\_\_poto\_\_ === Object.prototype

Object.prototype.\_\_proto\_\_ === null

1. 用过promise 封装原生ajax

let getJson = function(url){

let getUrl = new Promise(function(resolve,reject){

let change = function(){

if(this.readyState != 4 && this.stutas != 200){

resolve(this.response)

}else{

rejece(this.err)

}

}

let xmlHttp = new XmlHttpRequest();

xmlHttp.open('get',url,true);

xmlHttp.onreadystatechange = change;

xmlHttp.send();

})

return getUrl;

}

5 .http  与https

1. http 无状态 、 https(？)

2. https 加密 SSL ，

3.

6.tcp 握手

syn syn

ack = syn

1. vue mvvm 实现原理
2. 能否劫持数组的变化（）
3. vue如何实现对数组的监听
4. 为什么vue data返回是个函数

8、如何实现copy函数

1. 循环引用
2. function （如何处理）
3. 正则如何处理