

# 2022年前端面经分享

## 前言

分享我在2021年底到2022年初的面经，主攻前端方向和全栈方向。我是2021年10月份回国，之前在美国工作和创业了几年。回来之后发现国内和国外的面试方式有很大区别，面试会区分前后端，问很多八股文问题。刚开始准备的时候很烦，但是面多了会发现就那几个问题，而且有一定的套路，准备充分就不是问题了。

对于前端面试，React或者Vue的基础知识是必须掌握的，可以尝试手写一个简单的React。网上有教程，认真跟着写一遍之后，就相当于看过源代码了，对于React的八股文问题也会有很深的认识。其次是手写前端代码，比如手写bind, 手写防抖和限流，手写promise之类的问题。我汇总了我遇到的和网上总结的大概50个这样的问题，多练一练没错，遇到了不至于一脸懵逼。

非常抱歉，本文的顺序是倒着来的，上面的是最新，下面的是旧的。

[手写React](#) [手写前端代码高频题](#)

## 声网 - 现场 2,3,4 面

1. 后端工程师面试 - 考察前后端如何合作，前端部署流程的一些问题
2. HR面试 - 了解过往项目，创业经历
3. 负责人面试 - 了解过往项目，创业经历

## 声网 - 电面1面，前端工程师

箭头函数和普通函数的区别

箭头函数为什么不能new

let, var, const的区别是什么

electron线程之间如何通信？

JS事件循环是什么

react hook是什么

react hook如何unmount

实现一个 promise.allSettled，然后如何实现并发，但是按index顺序返回结果。

JS中如何判断是否是数组类型

call and apply 是什么？

箭头函数可以用call吗？为什么？

如何前端优化

webpack如何优化

## **Telsa - 电面5面，负责人面试**

介绍一个比较有挑战性的项目，我介绍了曾经开发过的身份验证系统

这个身份验证系统为什么使用 refresh token and access token 两个token，使用1个token不可以吗？然后负责人和我杠了起来

如何优化react

用英文介绍为什么想加入特斯拉

知道BFF吗？

## **Telsa - 电面4面，北美前端工程师面试**

What is your chanllenge as a CTO

What kind of candidate you want to hire

How you make a technical desicion on react native and electron

---

Why you choose react, and explain the advantage and disadvantage

Name some react hook and explain their functionalities

实现一个 reduce

算法题 objectAdd

## **Telsa - 电面3面，负责人面试**

问项目和过往经历

## **Tesla - 电面2面，前端工程师面试**

英文介绍项目

英文解释 react 虚拟DOM and fiber

React fiber

setState是同步还是异步

关于react hook的问题

关于react redux的问题

关于redux thunk的问题

什么是useMemo and useCallback，什么时候使用？

网络安全的问题，什么是XSS, CSRF

如何保护前端和后端的通信，HTTPS

## **Tesla - 电面1面，前端工程师面试**

英文介绍自己

---

关于 React Fiber 的问题

关于 React Hook 的问题

React 15 生命周期 componentDidMountReceiveProps, what is old props === new props

解释 DFS and BFS 的区别

解释 context and useRedux

解释 React Redux

解释 redux thunk 是什么

解释 Oauth2 的机制，因为简历中提到了，但是我忘记Oauth2的细节了

解释 CSS Flexbox

解释 CSS 选择器 > and +

## Tesla OA

1. 实现 react counter
2. 实现 react tabs
3. 实现 react hook 搜索框和防抖功能。react hook的搜索框防抖有别于react class防抖，是个坑需要注意

## Bayer - 电面1面，工程师面试

说说一些知道的设计模式，这些设计模式的使用场景

实现一个EventEmitter

实现一个深度克隆函数

实现异步加法 asyncAdd

```
// 假设有一台本地机器，无法做加减乘除运算（包括位运算），  
// 因此无法执行  $a + b$ 、 $a++ = 1$  这样的 JS 代码，
```

arduino 复制代码

```
// 然后我们提供一个服务器端的 HTTP API，可以传两个数字类型的参数，  
// 响应结果是这两个参数的和，这个 HTTP API 的 JS SDK（在本地机器上运行）的使用方法如下：
```

介绍项目

介绍前端移动端

介绍浏览器缓存策略

介绍后端相关问题

## SecondSpectrum OA

笔试题: 排列和统计一系列的股票交易数据，没搞明白

## 美团 - 电面1面，负责人面试

围绕最近一个项目的问题

关于项目管理

关于团队管理

一些技术上的问题，使用了什么技术和框架

介绍上一个项目的系统

一些大前端问题

一些后端问题

（岗位面高了，问了很多技术管理和技术框架上的问题）

## 上海人工智能实验室 - 电面1面，前端工程师面试

CSS margin left -10 会发生什么 right -10 会发生什么

CSS display none and display有什么区别

JS 箭头函数和普通函数的区别

instanceof 的原理是什么？原型链

箭头函数可以被new吗？不可以

new构造函数返回什么？如果函数的返回值是obj则返回这个obj，不然返回this。如果是1, 2, 3呢？返回this

构造函数bind之后，返回什么？返回绑定传入this的函数

REACT setState之后发生什么

REACT 15的更新机制

REACT 16的更新机制

React Hook是什么，为什么react hook

REACT hook useState的内部结构 - 链表

转换 list 成 tree 结构

ini 复制代码

```
const list = [
  { "id": 1, "pid": 0, "name": "部门a" },
  { "id": 2, "pid": 1, "name": "部门b" },
  { "id": 3, "pid": 1, "name": "部门c" },
  { "id": 4, "pid": 2, "name": "部门d" },
  { "id": 5, "pid": 3, "name": "部门e" },
  { "id": 6, "pid": 2, "name": "部门f" },
  { "id": 7, "pid": 3, "name": "部门g" },
  { "id": 8, "pid": 4, "name": "部门h" },
  { "id": 9, "pid": 4, "name": "部门i" },
  { "id": 10, "pid": 0, "name": "部门j" },
]
```

## 蔚来 - HR面

自我介绍

为什么回国

介绍上一个项目

当时的团队构成

未来职业规划

期望薪资

上一个工作的薪资

上一个的薪资构成

## 字节国际化电商 - 电面2面，前端工程师面试

介绍自己

介绍上一个项目

攻克过的技术难点

说一说曾经独立实现给哟的系统，说一说技术细节

算法题:

1. box-sizing (不会, 跳过)
2. 写一个react hook, 可以获得窗口的width和length, 支持resize
3. linked list intersection

为什么回国

如何和团队协作

node post请求中content类型有什么, 如何获得

node 中间件是什么, 如何封装一个中间件

说说如何全栈优化 - webpack, react, cdn, 缓存

---

说说http缓存策略

## 蔚来 - 电面3面，负责人面试

介绍背景

回国原因

有创业经历了为什么还会进大厂

对国内工作的了解

对国内公司的了解

为什么国内用H5开发和小程序开发

技术上的成就 - 搭建Auth系统

为什么这个系统中选择 indexedDB

创业团队的构成

实现一个深拷贝 - 包括 Date 的拷贝和循环引用的拷贝

## Jerry.ai OA

RangeList

## 蔚来 - 电面2面，前端工程师面试

React native 和原生的区别

React 的优势

CSS inline element 的问题

是否会 Regular expression, 不熟, 跳过

---



什么是JS中的宏任务和微任务

异步的打印顺序

算法题：小汽车加油

## 字节国际化电商 - 电面1面，前端工程师面试

设计搜索组件需要注意哪些，防抖和缓存

React 服务端渲染如何路由？这个问题我没听懂，问他他解释了10分钟我也不知道他要问啥，最后跳过了。

为什么使用React hook

React hook 为什么不能写在条件代码中

说说 Webpack 配置都有哪些

解释 webpack plugin and loader，都是什么，区别是什么

如何webpack优化

介绍上一个项目，使用服务端渲染

上一个项目中使用api signature，然后反复对这个地方提问

算法：Path to object

css 复制代码

```
const pathToObjData = {
  'a.b': 1,
  'a.c': 2,
  'a.d.e': 5,
  'c': 3
}
返回
{
  a: { b: 1, c: 2, d: { e: 5 } },
  c: 3
}
```

算法：Promise scheduler，实现一个最大并发数不能超过N的Promise限流器

## 蔚来 - 电面1面，前端工程师面试

介绍项目

什么是webpack loader and plugin，区别和使用场景

说说 react VDOM and DOM是什么，为什么？

算法1 jsonToString，手写一个 JSON.stringify 的方法

算法2 treeToList

yaml 复制代码

```
/*
 *以下数据结构中，id 代表部门编号，name 代表部门名称，parentId 是父级部门编号，parentId为 0 代表一级部门
 * 现在要求：
 1: 把树形结构转换成原始list。
 2:把原始list重新还原成树形结构。
 * 加分项：尽可能降低时间复杂度。
 */
// 原始 树形结构 如下
let tree = [
  {
    id: 1,
    name: '部门A',
    parentId: 0,
    children: [
      {
        id: 3,
        name: '部门C',
        parentId: 1,
        children: [
          {
            id: 6,
            name: '部门F',
            parentId: 3
          }
        ]
      }
    ],
  },
  {
    id: 4,
    name: '部门D',
    parentId: 1,
    children: [
```

```
    {
      id: 8,
      name: '部门H',
      parentId: 4
    }
  ]
}
]
},
{
  id: 2,
  name: '部门B',
  parentId: 0,
  children: [
    {
      id: 5,
      name: '部门E',
      parentId: 2
    },
    {
      id: 7,
      name: '部门G',
      parentId: 2
    }
  ]
}
];
```

转换后的结果

```
let list = [
  {id:1, name:'部门A', parentId:0},
  {id:2, name:'部门B', parentId:0},
  {id:3, name:'部门C', parentId:1},
  {id:4, name:'部门D', parentId:1},
  {id:5, name:'部门E', parentId:2},
  {id:6, name:'部门F', parentId:3},
  {id:7, name:'部门G', parentId:2},
  {id:8, name:'部门H', parentId:4}
];
```

## 阿里巴巴本地生活 - 电面1面，负责人面试

关于 React Native 的问题

CSS布局，如何左右固定100px宽，中间自适应。说出来的方法越多越好

RN下拉菜单，如何在下方显示菜单，如何关闭

什么是RN中的JS bridge

如何RN Webview 中获得 callback

JS event, current target和什么target的区别

异步操作的顺序

promise 设计超时逻辑

介绍我的项目中的服务端渲染

介绍低代码

## Coupang - 电面2面，工程师面试

1. 现场使用 RN 实现一个微信聊天界面
2. 如何团队合作的，技术上的
3. 实现一个timestampToDate
4. 设计模式的问题，如何设计一个abcd buttons互斥的逻辑

## 字节跳动 - 电面1面，工程师面试

http有什么请求头控制cache

react key是什么，为什么需要key

react hook 为什么不能写在条件代码中

HTML如何判断用户进入可视范围

实现一个 Promise.race

实现一个 Promise concurrency control，控制最大并发数

JS中for of 和 for in的区别

---

什么是 symbol.iterator

讨论了一下 React + H5

## **哔哩哔哩 - 电面1面，负责人面试**

介绍上一个项目

如果弹幕太多，如何去重

关于团队管理的问题

如何和前端工程师合作

## **Coupang - 电面1面，负责人面试**

介绍过往经历，项目背景

自己的优势和弱势

遇到的困难

围绕前端框架的一些问题

还有一些后端框架的问题

是否了解microservice

是否会java

## **Shopee OA**

不知道为什么给我的后端岗位

10个选择题，都是后端相关的，数据库，事务锁

2个算法，但是在线编程网站需要自己调试输入和输出，搞了半个小时

---

## 1. 合并2个有序数组成一个有序数组

ini 复制代码

```
var readline = require('readline');
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});

let input = 0
let nums1 = []
let nums2 = []
rl.on('line', function(line){
  if (input === 0) {
    nums1 = line.split(',');
    input++
  } else if (input === 1) {
    nums2 = line.split(',');
    input++
  }
  if (input === 2) {
    let res = merge(nums1, nums2)
    console.log(res);
    rl.close();
  }
});

var merge = function(nums1, nums2) {
  if (typeof nums1 === 'string') nums1 = nums1.join(',')
  if (typeof nums2 === 'string') nums2 = nums2.join(',')

  let n1 = nums1.length
  let n2 = nums2.length
  let res = []
  let index1 = 0
  let index2 = 0

  while (index1 < n1 || index2 < n2) {
    let num1 = parseInt(nums1[index1])
    let num2 = parseInt(nums2[index2])
    if (index1 < n1 && index2 < n2 && num1 < num2 || index1 < n1 && index2 >= n2) {
      res.push(nums1[index1])
      index1++
    } else {
      res.push(nums2[index2])
      index2++
    }
  }

  return res.join(',')
}
```

```
}
```

## 2. 通过前序遍历和后序遍历，恢复一个2查树

ini 复制代码

```
var readline = require('readline');
const rl = readline.createInterface({
  input: process.stdin,
  output: process.stdout
});

let input = 0
let n = 0
let inorder = []
let postorder = []
rl.on('line', function(line){
  if (input === 0) {
    n = line
    input++
  } else if (input === 1) {
    inorder = line.split(' ');
    input++
  } else if (input === 2) {
    postorder = line.split(' ');
    input++
  }
  if (input === 3) {
    let root = createTree(inorder, postorder)
    let res = []
    output(root, res)
    console.log(res.join(','));
    rl.close();
  }
});

function TreeNode(val, left, right) {
  this.val = (val === undefined ? 0 : val)
  this.left = (left === undefined ? 0 : left)
  this.right = (right === undefined ? 0 : right)
}

var createTree = function(inorder, postorder) {
  let index = postorder.length - 1
  let map = new Map()
  for (let i = 0; i < inorder.length; i++) {
    map.set(inorder[i], i)
  }
```

```
var dfs = function(inLeft, inRight) {
    if (inLeft > inRight || index < 0) {
        return null
    }

    let key = postorder[index]
    let node = new TreeNode(key)
    let mid = map.get(key)
    index = index - 1

    node.right = dfs(mid + 1, inRight)
    node.left = dfs(inLeft, mid - 1)
    return node
}

return dfs(0, inorder.length - 1)
}

var output = function(root, res) {
    if (!root) return null
    res.push(root.val)
    output(root.left, res)
    output(root.right, res)
}
```

## Grab - 电面1面，负责人面试

介绍过往经历和项目背景

## Authing - 现场2面

### 前端工程师

用react实现一个modal，并且完成一些功能

1. 可以开关
2. modal模块化
3. 实现 modal 动画效果，css transition

### 后端工程师



数据库问题, nodejs, redis, 异步问题, 高并发问题, 算法问题:

1. 搜索一个数组中的range
2. 全排列

## **PtMind - 电面1面, 负责人面试**

简历中提到的后端安全问题是什么? 解释了一下

项目中遇到了什么挑战?

技术决策问题, 如何选择一个开源项目

如何对前端的优化

如何优化后端

NODE多线程之间如何通信 - IPC

当时系统的QPS是多少?

解释了Next SSR如何解决SEO的问题

Vue和react的区别?

如何计算开发时间?

使用过AWS吗?

上一个项目的前端UIUX设计是如何做的?

## **Crimson Education - 电面2面**

**后端工程师面试:**

用户输入URL之后发生了什么?

关于Node的问题

---

什么是 node event loop

CDN的原理

关于 Nginx and load balancer 的问题

什么是 IPC

electron 是什么框架?

## **前端工程师面试:**

react 虚拟dom 会更快吗?

不用react eject如何进行配置?

componentWillReceiveProps 的机制

事件冒泡机制

关于React hook的问题

useEffect, useMemo, useReducer, context都是什么

什么是 JS 原型链

什么是 Async/Await

浏览器的机制

缓存机制

## **负责人面试 (全程英语)**

技术决策相关

团队合作相关

项目管理相关

---

如何和团队决策，过程是什么？

如果有一个新的需求，接下来你要做什么？

决策如何做技术文档？使用什么工具？

## Authing - 电面1面，CTO面试

介绍上个项目

前端做过的框架，解决的问题

后端做过的框架，解决的问题

介绍上上个项目

前端是如何实现的，解决的问题

## Crimson Education OA

1. firstNonRepeatingLetter
2. Multiples of 3 and 5
3. LookSay
4. Design a pagination class
5. change coins (Dynamic Programming)
6. SQL - group by and order by date, group by and count description

```
select date(created_at) as day, description, count(*) as count from events
where name = 'trained' group by day, description order by day asc
```

sql 复制代码

## Crimson Education - 电面1面，负责人面试

项目背景，过往经历的一些问题

## LeetCode - 电面1面，工程师面试

什么使服务端渲染

什么是 react hydrate

如何技术选型，会看源代码吗？

HTTP/HTTPS的问题，HTTPS的机制

TCP/UDP的区别，特性

了解webRTC吗？

NODE和浏览器事件循环

手写一个reduce

css问题，header置顶，footer在最下面，中间部分为内容展示部分。即使中间内容为空，footer也保持最下面

## 字节广告组 - 电面1面，工程师面试

上来介绍了他们组的业务方向

是否愿意做解决方案工程师

自我介绍

介绍上个项目

算法 ['java', 'python', 'go'] 每个字母的全部组合，和leetcode #17 类似，电话号码，全排列

## 小红书 - 电面1面，前端工程师面试

React Next的原理

React的更新原理

React hook是什么，原理是什么

---

React Fiber是什么，原理是什么

Redux 原理是什么

什么是 XSS 和 CSRF

算法 - N叉树寻找目标路径

## 思谋 - 电面1面，前端工程师面试

react的更新原理

react fiber的原理

react key是什么？

CSS的问题

异步方法的输出顺序

scss 复制代码

```
console.log(1) // 1 同步

setTimeout(() => {
  console.log(2) // 7. 任务队列, timer 队列
}, 0)

setTimeout(() => {
  console.log(3) // 8. 任务队列, timer 队列
}, 0)

Promise.resolve()
  .then(function() {
    console.log(4) // 4. 循环1
  })
  .then(function() {
    console.log(5) // 6 循环3
  })

async function foo() {
  await bar() // 2 同步
  console.log(6) // 5 循环2
}

foo()
```

```
function bar() {  
  console.log(7)  
}  
console.log(8) // 3 同步  
  
/**  
同步  
1  
7  
8  
  
promise 比 await下面的优先  
4  
6  
5  
  
set timeout 优先最低  
2  
3  
*/
```