# **5年前端 历时1个月收获7个offer**

<https://juejin.cn/post/7142690757722243102>

**省流**：最终拿到了58、UMU、便利蜂、虾皮、快手、腾讯、字节的offer。

金三银四面试的, 这次整体面试通过率还挺高的, 面试前没有太多准备, 基本上是面试过程中不断复盘, 整理面试遇到的问题, 到最后面几家自己心仪的公司, 其实就会发现大家问的问题都差不多。

时间过去比较久了, 最近得空了稍微整理一下发出来(不过有些比较重复的问题我当时没有记录, 现在也记不起来了)

## 滴滴

第一家就面的滴滴, 啥都没准备, 中间很多没答好, 但是意外的到了HR面, 可能由于面试表现并不好, 所以给的薪资不及预期。

其中印象比较深刻的是三面面试官：

面试官：问A入职后和上级意见不合应该怎么处理  
我：我官话回答了半天说要考虑当时的背景、双方的观点正确与否再考虑, 最终选择最有利于业务发展的一方  
面试官：说这些都没用, 如果最终上司的方案确实不如A的, 但上司就是坚持自己的意见怎么办?  
我：那我不知道, 请问您有什么看法  
面试官：不招A就行了, 面试阶段就不能让他通过

### 一面

* 闭包是什么? 闭包的用途?
* 简述事件循环原理
* 虚拟dom是什么? 原理? 优缺点?
* vue 和 react 在虚拟dom的diff上，做了哪些改进使得速度很快?
* vue 和 react 里的key的作用是什么? 为什么不能用Index？用了会怎样? 如果不加key会怎样?
* vue 双向绑定的原理是什么?
* vue 的keep-alive的作用是什么？怎么实现的？如何刷新的?
* vue 是怎么解析template的? template会变成什么?
* 如何解析指令? 模板变量? html标签
* 用过vue 的render吗? render和template有什么关系

【代码题】 实现一个节流函数? 如果想要最后一次必须执行的话怎么实现?

【代码题】 实现一个批量请求函数, 能够限制并发量?

### 二面

【代码题】 数组转树结构

const arr = [{

id: 2,

name: '部门B',

parentId: 0

},

{

id: 3,

name: '部门C',

parentId: 1

},

{

id: 1,

name: '部门A',

parentId: 2

},

{

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

}

]复制代码

### 终面

【代码题】 去除字符串中出现次数最少的字符，不改变原字符串的顺序。

“ababac” —— “ababa”

“aaabbbcceeff” —— “aaabbb”复制代码

【代码题】 写出一个函数trans，将数字转换成汉语的输出，输入为不超过10000亿的数字。

trans(123456) —— 十二万三千四百五十六

trans（100010001）—— 一亿零一万零一复制代码

## 58 (offer)

整体面试比较顺利, 就是没想到三轮远程面试后, 最终还去现场经历了一次交叉面和业务负责人面试, 不过HR确实是很热情也很专业。不过最终选择了其他offer, 甚至有点感觉对不起大家的热情。

一面二面三面都很不错, 交叉面和业务负责人面试有点水, 就随便问问。

### 一面

* 对前端工程化的理解
* 前端性能优化都做了哪些工作
* Nodejs 异步IO模型
* libuv
* 设计模式
* 微前端
* 节流和防抖
* react有自己封装一些自定义hooks吗? vue有自己封装一些指令吗
* vue 向 react迁移是怎么做的? 怎么保证兼容的
* vue的双向绑定原理
* Node的日志和负载均衡怎么做的
* 那前后端的分工是怎样的？哪些后端做哪些node做
* 给出代码的输出顺序

async function async1() {

console.log('async1 start');

await async2();

console.log('async1 end');

}async function async2() {

console.log('async2');

}console.log('script start');setTimeout(function () {

console.log('setTimeout');

}, 0)async1();new Promise(function (resolve) {

console.log('promise1');

resolve();

console.log('promise2')

}).then(function () {

console.log('promise3');

});console.log('script end');复制代码

【代码题】 给几个数组, 可以通过数值找到对应的数组名称

// 比如这个函数输入一个1，那么要求函数返回Aconst A = [1,2,3];const B = [4,5,6];const C = [7,8,9];

function test(num) {

}复制代码

### 二面

* 了解过vue3吗?
* webscoket的连接原理
* react生命周期
* redux原理
* vue 父子组件的通信方式
* async await的原理是什么?
* async/await, generator, promise这三者的关联和区别是什么?

【场景设计】 设计一个转盘组件, 需要考虑什么, 需要和业务方协调好哪些技术细节? 前端如何防刷

### 三面

【代码题】 数组转树, 写完后问如果要在树中新增节点或者删除节点, 函数应该怎么扩展

const arr = [{

id: 2,

name: '部门B',

parentId: 0

},

{

id: 3,

name: '部门C',

parentId: 1

},

{

id: 1,

name: '部门A',

parentId: 2

},

{

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

}

]复制代码

### 交叉面

* 虚拟列表怎么实现?
* 做过哪些性能优化?

### 终面

* 都是一些项目相关

## 金山

一面感觉不错, 面试官非常专业, 态度也和蔼可亲;  
终面的大哥比较盛气凌人, 疯狂PUA, 聊完后让我降薪, 就直接告辞了。

### 一面

* react和vue在技术层面的区别
* 常用的hook都有哪些?
* 用hook都遇到过哪些坑?
* 了解useReducer吗
* 组件外侧let a 1 组件内侧点击事件更改a，渲染的a会发生改变吗？如果let a放在组件内部，有什么变化吗？和useState有什么区别？
* 了解过vue3吗?
* Node是怎么部署的? pm2守护进程的原理?
* Node开启子进程的方法有哪些?
* 进程间如何通信?
* css 三列等宽布局如何实现? flex 1是代表什么意思？分别有哪些属性?
* 前端安全都了解哪些? xss csrf
  + csp是为了解决什么问题的?
  + https是如何安全通信的?
* 前端性能优化做了哪些工作?

【代码题】 不定长二维数组的全排列

// 输入 [['A', 'B', ...], [1, 2], ['a', 'b'], ...]

// 输出 ['A1a', 'A1b', ....]复制代码

【代码题】 两个字符串对比, 得出结论都做了什么操作, 比如插入或者删除

pre = 'abcde123'

now = '1abc123'

a前面插入了1, c后面删除了de复制代码

### 终面

【场景设计】 大数据列表如何设计平滑滚动和加载，下滑再上滑的操作，上下两个buffer区间如何变化和加载数据。

## 便利蜂 (offer)

整体面试比较顺利, 三位面试官也都比较健谈, 最终给了一个很高的总包。不过感觉面试题太简单, 给的钱又多, 有点担心就选择了其他offer。

### 一面

纯聊项目

### 二面

* js中的闭包
* 解决过的一些线上问题
* 线上监控 对于crashed这种怎么监控? 对于内存持续增长，比如用了15分钟之后才会出现问题怎么监控
* 对于linux熟吗? top命令的属性大概聊一下?
* 301 302 304的区别

### 三面

【代码题】 sleep函数

【代码题】 节流防抖

## 小红书

整体给我的感觉是为了面试而面试, 体验极差。

一面面试官只是机械的提问, 提问完也不认真听我的回答, 上一个问题跟下一个问题根本没有关联性, 就像是在对着题库随便选题。

二面面试时好像一直在电脑上聊天, 结束后说是会约三面, 过了大概两周说是只招leader, 我不符合。

### 一面

* 输出什么? 为什么?

var b = 10;

(function b(){

b = 20;

console.log(b);

})();复制代码

* 代码输出顺序题

async function async1() {

console.log('1');

await async2();

console.log('2');

}

async function async2() {

console.log('3');

}

console.log('4');

setTimeout(function() {

console.log('5');

}, 0);

async1();

new Promise(function(resolve) {

console.log('6');

resolve();

}).then(function() {

console.log('7');

});

console.log('8');复制代码

* async await的原理是什么?
* async/await, generator, promise这三者的关联和区别是什么?
* BFC是什么? 哪些属性可以构成一个BFC呢?
* postion属性大概讲一下, static是什么表现? static在文档流里吗?
* Webpack的原理, plugin loader 热更新等等
* Set和Map
* vue的keep-alive原理以及生命周期
* vuex

【代码题】 ES5和ES6的继承? 这两种方式除了写法, 还有其他区别吗?

【代码题】 EventEmitter

### 二面

* 浏览器从输入url开始发生了什么
* react生命周期
* redux的原理
* vue 父子组件的通信方式
* vue的双向绑定原理
* 对vue3的了解? vue3是怎么做的双向绑定?

【代码题】 使用Promise实现一个异步流量控制的函数, 比如一共10个请求, 每个请求的快慢不同, 最多同时3个请求发出, 快的一个请求返回后, 就从剩下的7个请求里再找一个放进请求池里, 如此循环。

## UMU (offer)

前两面都是远程, 终面去了公司现场。到了之后先做了一份测试题, 大概就是考察基本认知能力的。完事后CTO来面试, 直接在现场小黑板上写题。大佬对各种技术都能侃侃而谈, 确实很厉害, 不过听说周六可能要加班, 而且担心稳定性, 就没选择这里。

### 一面

除了项目, 基本都是问的日常开发相关的东西, 比较实在

* node.js如何调试
* charles map local/map remote
* chrome devtool 如何查看内存情况

### 二面

* koa洋葱模型
* 中间件的异常处理是怎么做的？
* 在没有async await 的时候, koa是怎么实现的洋葱模型?
* body-parser 中间件了解过吗
* 如果浏览器端用post接口上传图片和一些其他字段, header里会有什么? koa里如果不用body-parser，应该怎么解析?
* webscoket的连接原理
* https是如何保证安全的? 是如何保证不被中间人攻击的?

### 终面

【代码题】 给一个字符串, 找到第一个不重复的字符

ababcbdsa

abcdefg复制代码

* 时间复杂度是多少?
* 除了给的两个用例, 还能想到什么用例来测试一下?

## 网易

一面直接挂掉, 代码题整体写的乱七八糟, 挂掉理所应当...

### 一面

* 你觉得js里this的设计怎么样? 有没有什么缺点啥的
* vue的响应式开发比命令式有什么好处
* 装饰器
* vuex
* generator 是如何做到中断和恢复的
* function 和 箭头函数的定义有什么区别? 导致了在this指向这块表现不同
* 导致js里this指向混乱的原因是什么?
* 浏览器的事件循环
* 宏任务和微任务的区分是为了做什么? 优先级?

【代码题】 实现compose函数, 类似于koa的中间件洋葱模型

// 题目需求

let middleware = []

middleware.push((next) => {

console.log(1)

next()

console.log(1.1)

})

middleware.push((next) => {

console.log(2)

next()

console.log(2.1)

})

middleware.push((next) => {

console.log(3)

next()

console.log(3.1)

})

let fn = compose(middleware)fn()

/\*

1

2

3

3.1

2.1

1.1

\*/

//实现compose函数function compose(middlewares) {

}

复制代码

【代码题】 遇到退格字符就删除前面的字符, 遇到两个退格就删除两个字符

// 比较含有退格的字符串，"<-"代表退格键，"<"和"-"均为正常字符// 输入："a<-b<-", "c<-d<-"，结果：true，解释：都为""// 输入："<-<-ab<-", "<-<-<-<-a"，结果：true，解释：都为"a"// 输入："<-<ab<-c", "<<-<a<-<-c"，结果：false，解释："<ac" !== "c"

function fn(str1, str2) {

}

复制代码

## 快手 (offer)

整体面试的感觉很专业, 面试态度也很认真, 考察的比较全面, 不过比较蛋疼的是HR面结束后拖了好久, 我Shopee、腾讯、字节口头offer都拿到了, 快手最后才给的口头offer, 也可能是想对比一下其他家的价格吧。

### 一面

* 小程序的架构? 双线程分别做的什么事情?
* 为什么小程序里拿不到dom相关的api
* 代码输出题

console.log(typeof typeof typeof null);console.log(typeof console.log(1));复制代码

* this指向题

var name = '123';

var obj = {

name: '456',

print: function() {

function a() {

console.log(this.name);

}

a();

}

}

obj.print();复制代码

【代码题】 实现一个函数, 可以间隔输出

function createRepeat(fn, repeat, interval) {}

const fn = createRepeat(console.log, 3, 4);

fn('helloWorld'); // 每4秒输出一次helloWorld, 输出3次复制代码

【代码题】 删除链表的一个节点

function (head, node) {}复制代码

【代码题】 实现LRU算法

class LRU {

get(key) {

}

set(key, value) {

}

}复制代码

### 二面

* Promise then 第二个参数和catch的区别是什么?
* Promise finally 怎么实现的
* 作用域链
* Electron架构
* 微前端
* webpack5 模块联邦
* Webworker

### 三面

没有记录, 应该都是问的项目

## 高德

一面直接挂掉, 面试官一直怼着各种api考察记忆力, 都是一些非常偏非常不常用的api。

* 比如css有没有用过xx属性, 是做什么的?
* Promise有个xx方法, 你知道是做什么的吗?

最终就是挂在了css的各种属性背诵上, 麻了, 面试的时候让候选人从头背这些api真的有意义吗?

### 一面

* Symbol
* useRef / ref / forwardsRef 的区别是什么?
* useEffect的第二个参数, 传空数组和传依赖数组有什么区别?
  + 如果return 了一个函数, 传空数组的话是在什么时候执行? 传依赖数组的时候是在什么时候执行?
* flex布局
* ES5继承
* ES6继承, 静态方法/属性和实例方法/属性 是什么时候挂载的
* Promise各种api
* 各种css属性

## Shopee (offer)

面试比较痛快, 一共两面。一下午直接搞定, 一面结束后直接约了一个小时后二面, 二面结束后直接约了一个小时后的HR面。但是最后谈薪的时候HR也是比较磨叽, 一直说要等快手或者字节的价格出来, 他们才会给价格, 说是这样才更有竞争力。

### 一面

* 各种缓存的优先级, memory disk http2 push?
* 小程序为什么会有两个线程? 怎么设计?
* xss? 如何防范?
* 日志 如果大量日志堆在内存里怎么办?

【代码题】 深拷贝

const deepClone = (obj, m) => {

};

需要手写一个深拷贝函数deepClone，输入可以是任意JS数据类型复制代码

【代码题】 二叉树光照，输出能被光照到的节点, dfs能否解决?

输入: [1,2,3,null,5,null,4]

输出: [1,3,4]

输入: []

输出: []

/\*\*

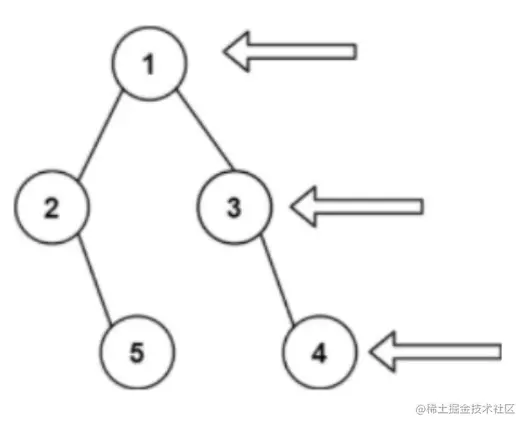
\* @param {TreeNode} root

\* @return {number[]}

\*/function exposedElement(root) {

};

复制代码



* **【代码题】** 输出顺序题

setTimeout(function () {

console.log(1);

}, 100);

new Promise(function (resolve) {

console.log(2);

resolve();

console.log(3);

}).then(function () {

console.log(4);

new Promise((resove, reject) => {

console.log(5);

setTimeout(() => {

console.log(6);

}, 10);

})

});console.log(7);console.log(8);复制代码

【代码题】 作用域

var a=3;

function c(){

alert(a);

}

(function(){

var a=4;

c();

})();复制代码

【代码题】 输出题

function Foo(){

Foo.a = function(){

console.log(1);

}

this.a = function(){

console.log(2)

}

}

Foo.prototype.a = function(){

console.log(3);

}

Foo.a = function(){

console.log(4);

}

Foo.a();let obj = new Foo();

obj.a();Foo.a();复制代码

### 终面

* 错误捕捉
* 前端稳定性监控
* 前端内存处理

【代码题】 好多请求, 耗时不同, 按照顺序输出, 尽可能保证快, 写一个函数.

const promiseList = [

new Promise((resolve) => {

setTimeout(resolve, 1000)

}),

new Promise((resolve) => {

setTimeout(resolve, 1000)

}),

new Promise((resolve) => {

setTimeout(resolve, 1000)

})

]

fn(promiseList);复制代码

【代码题】 一个数组的全排列

输入一个数组 arr = [1,2,3]

输出全排列

[[1], [2], [3], [1, 2], [1, 3], ...., [1,2,3], [1,2,4] ....]复制代码

## 腾讯 (offer)

腾讯的整体流程是最长的, 一共面了5次, 整体面试难度倒不高, 只有前三轮是正经问技术问题, 后面更像是在聊天聊项目。

因为当时整体其他offer都快到ddl了, 就赶紧催hr给个数字, 最终薪资不及预期就没去了。

### 一面

* 普通函数和箭头函数的this指向问题

const obj = {

fn1: () => console.log(this),

fn2: function() {console.log(this)}

}

obj.fn1();

obj.fn2();

const x = new obj.fn1();const y = new obj.fn2();复制代码

* promise相关的特性
* vue父子组件, 生命周期执行顺序 created mounted
* vue3添加了哪些新特性?
* xss 的特点以及如何防范?
* Http 2.0和http3.0对比之前的版本, 分别做了哪些改进?
* HTTP 3.0基于udp的话, 如何保证可靠的传输?
* TCP和UDP最大的区别是什么?
* CSP除了能防止加载外域脚本, 还能做什么?
* typescript is这个关键字是做什么呢?

【代码题】 多叉树, 获取每一层的节点之和

function layerSum(root) {

}

const res = layerSum({

value: 2,

children: [

{ value: 6, children: [ { value: 1 } ] },

{ value: 3, children: [ { value: 2 }, { value: 3 }, { value: 4 } ] },

{ value: 5, children: [ { value: 7 }, { value: 8 } ] }

]

});

console.log(res);

复制代码

### 二面

没记录, 应该是和前面遇到的面试题重复了

【代码题】 虚拟dom转真实dom

const vnode = {

tag: 'DIV',

attrs: {

id: 'app'

},

children: [{

tag: 'SPAN',

children: [{

tag: 'A',

children: []

}]

},

{

tag: 'SPAN',

children: [{

tag: 'A',

children: []

},

{

tag: 'A',

children: []

}

]

}

]

}

function render(vnode) {

}复制代码

### 三面

* 前端安全 xss之类的
* Https中间人攻击
* 前端History路由配置 nginx

【代码题】 有序数组原地去重

### 四面(gm?忘记了)

如何估算一个城市中的井盖数量

### 终面

都是项目

## 字节 (offer)

之前听说字节各种hard, 给我吓惨了, 不过可能是运气好, 碰到的题都很简单。

### 一面

【代码题】 二叉树层序遍历, 每层的节点放到一个数组里

给定一个二叉树，返回该二叉树层序遍历的结果，（从左到右，一层一层地遍历）

例如：

给定的二叉树是{3,9,20,#,#,15,7},

该二叉树层序遍历的结果是 [ [3], [9,20], [15,7]

]复制代码

【代码题】 实现一个函数, fetchWithRetry 会自动重试3次，任意一次成功直接返回

【代码题】 链表中环的入口节点

对于一个给定的链表，返回环的入口节点，如果没有环，返回null复制代码

### 二面

* 截图怎么实现
* qps达到峰值了，怎么去优化
* 谷歌图片, 如果要实现一个类似的系统或者页面, 你会怎么做?
* 最小的k个数
* 节流防抖
* sleep函数
* js超过Number最大值的数怎么处理?
* 64个运动员, 8个跑道, 如果要选出前四名, 至少跑几次?
* 前端路由 a -> b -> c这样前进, 也可以返回 c -> b -> a, 用什么数据结构来存比较高效
* node 服务治理

【代码题】 多叉树指定层节点的个数

### 三面

【代码题】 叠词的数量

Input: 'abcdaaabbccccdddefgaaa'Output: 4

1. 输出叠词的数量2. 输出去重叠词的数量3. 用正则实现复制代码

## 写在后面

最近的整体行情好像更差了, Shopee也传出了很多毁offer的事情, 当时幸亏没去...

不过劝想看机会的大伙还是骑驴找马, 可以先简单试试水, 看看行情到底如何, 不要只听网友怎么说。

最终愿大家变成offer收割机, 每个人都能拿到自己心仪的offer。

链接：https://juejin.cn/post/7142690757722243102  
来源：稀土掘金  
著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。