# **准备大纲**----------------------

1.算法 ：排序 查找 数据结构

2.基础：H5,CSS,JS,JQuery, HTTP响应码,三次握手，四次分手

3.项目经验：

1.多源下载：html css js jquery ajax php mysql

难点：进度条，前台显示。

2.http反向代理：（回顾一下）

难点：

3.零食系统：html css js jquery（有支付 就好了！）

4.在线考试系统： 难点：ajax 算分，组卷。（在线考试、自主测试），科目。

5.智能作业： 和上面差不多（班级，学生，老师，分数）

6.人员管理：增删改查

7.纯前台

4.自己博客上的东西

5.面试题

## 1.算法准备--------------------------------

排序：插入、选择、冒泡、快排。

排序：内部排序和外部排序。

内部排序（8）：插入排序：直接插入排序 希尔排序

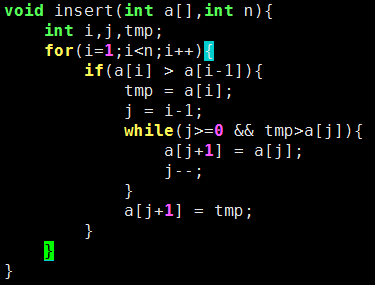
选择排序：简单选择排序 堆排序 O(nlog2n)

交换排序：冒泡排序 快速排序 O(nlog2n)

归并排序 O(nlog2n)

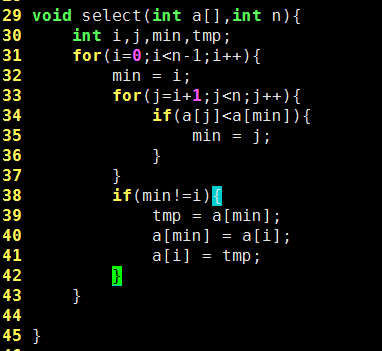
基数排序

直接插入排序（InsertSort）：稳定的，因为相等的还在原来顺序的后面。O（n^2）



简单选择排序（selectSort）：在要排序的一组数中，选出最小（或者最大）的一个数与第1个位置的数交换；然后在剩下的数当中再找最小（或者最大）的与第2个位置的数交换，依次类推，直到第n-1个元素（倒数第二个数）和第n个元素（最后一个数）比较为止。O（n^2）

//选择排序



void select\_sort(int\*a,int n)

{

register int i,j,min,t;

for(i=0;i<n-1;i++)

{

min=i;//查找最小值

for(j=i+1;j<n;j++)

if(a[min]>a[j])

min=j;//交换

if(min!=i)

{

t=a[min];

a[min]=a[i];

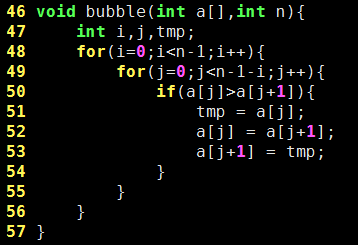
a[i]=t;

}

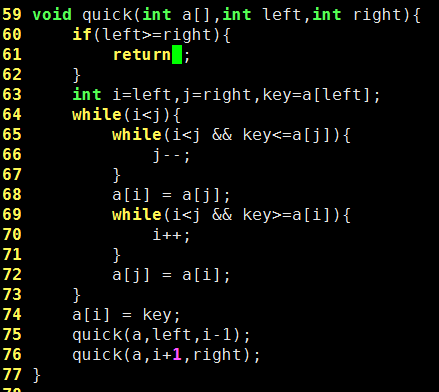
}

}

二元选择排序（good！）：改进为每趟循环确定两个元素（当前趟最大和最小记录）的位置,从而减少排序所需的循环次数。改进后对n个数据进行排序，最多只需进行[n/2]趟循环即可。 O（n^2）

冒泡排序（bubbleSort）：j 和j+1比较， 

快排（quickSort）：



void sort(int \*a, int left, int right)

{

if(left >= right)

{

return ;

}

int i = left;

int j = right;

int key = a[left];

while(i < j)

{

while(i < j && key <= a[j])

{

j--;/\*向前寻找\*/

}

a[i] = a[j];

while(i < j && key >= a[i])

{

i++;

}

a[j] = a[i];

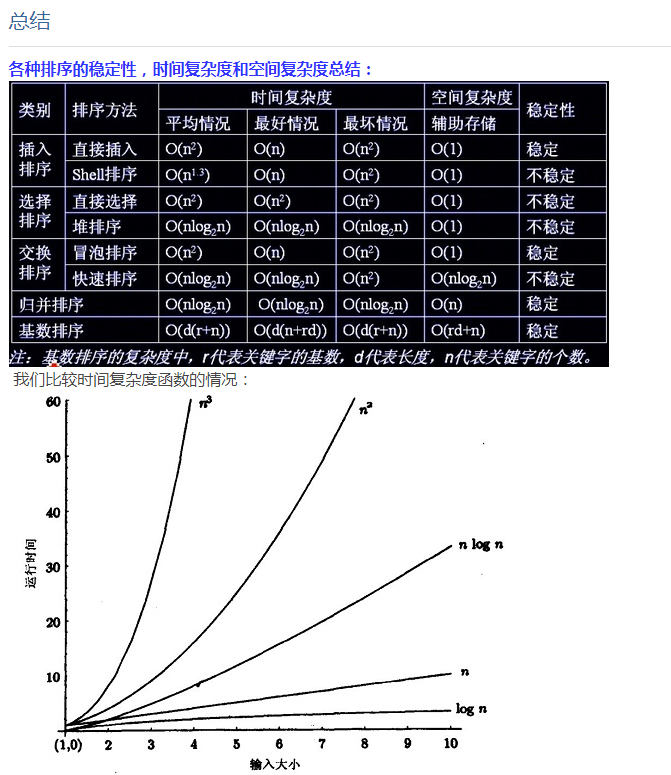
}

a[i] = key;

sort(a, left, i - 1);

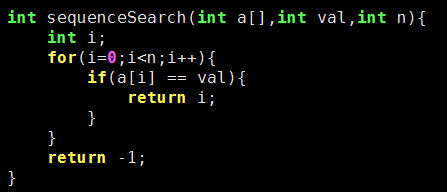
sort(a, i + 1, right);

}



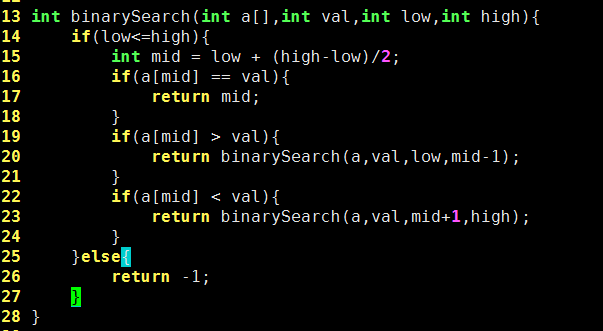
查找（7）：顺序查找，二分查找，插值查找，斐波那契查找，树表查找，分块查找，哈希查找

顺序查找：O（n）；

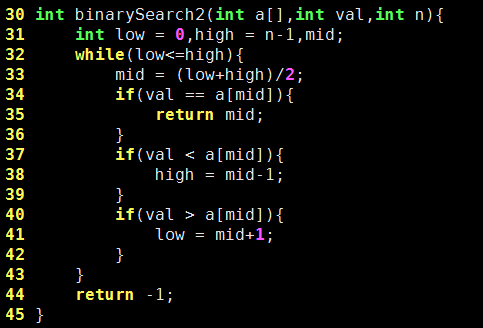


二分查找（折半查找）：数组必须是有序的。O(logn)

1.递归的



2.非递归的



## 2.面试题------------------------------------------------

作者：猪队  
链接：<https://www.nowcoder.com/discuss/59663?type=2&order=0&pos=34&page=1>  
来源：牛客网  
  
面试前笔试  
深复制  
数组去重  
String.prototype.replace的用法  
  
一面  
vue双向数据绑定  
前端性能优化  
如何测试首屏加载的时间（在文档开始处以及首屏末尾处添加一个时间戳，相减为首屏加载时间）  
css盒子模型  
es6有哪些新特性  
let和const的区别  
如何优化弹幕  
  
**腾讯**  
一面 技术面  
快排  
堆排序  
红黑树  
DNS查询过程  
负载均衡  
nginx 反向代理  
一致性哈希算法  
XSS  
CSRF  
DNS劫持  
RSA算法  
浏览器缓存  
如何让浏览器不缓存  
前端优化  
常见HTTP状态码有哪些，重点问了503遇到过吗（当我想说100那些时，面试官说不用那些，你只用说常见的就行了）  
es6有哪些新特性  
讲讲arraybuffer  
讲讲异步解决方案  
es2017的async/await实现原理  
常见的设计模式有哪些  
简单聊了下项目  
项目遇到的困难以及如何解决的  
有什么问题想问我吗  
  
  
二面 总监面（面试官比较严肃）  
讲讲项目用到哪些技术  
css有什么用  
你用过哪些css技术  
讲讲js有哪些技术  
js有哪些接口，有哪些函数  
你知道哪些排序算法  
来写个归并排序吧  
有没有什么问题想问我  
  
京东  
一面  
get和post的区别  
vue双向数据绑定（以及如何渲染的，虚拟dom）  
如何让一个不定宽高的元素水平垂直居中  
promise和setTimeout谁会优先执行  
跨域有哪些方法  
讲一讲CORS  
手写jsonp  
手写洗牌算法  
  
**微众**  
一面（面试官是架构师）  
angular和vue有什么区别  
vue,angular双向绑定  
mvvm，其中vm是什么，有什么作用  
项目（拿一个项目问，问的比较详细）  
同源策略  
跨域有哪些方法  
如何阻止别人的跨域访问  
  
二面（领导面）  
做过的项目（主要是问项目）  
以前的公司  
为什么不留以前的公司而想来微众  
为什么想做前端  
  
hr面  
自我介绍  
最近压力大吗  
有别的公司offer吗  
你认为没拿到别的公司的offer的原因是什么  
如果微众没有给到你offer怎么办  
最近令你最沮丧的一件事是什么  
令你最有成就感的一件事  
家庭情况  
学校发三方了吗

## 3.http响应码------------------------------------------

**http状态返回代码 1xx（临时响应）**  
表示临时响应并需要请求者继续执行操作的状态代码。

**http状态返回代码** 代码   说明  
100   （继续） 请求者应当继续提出请求。 服务器返回此代码表示已收到请求的第一部分，正在等待其余部分。   
101   （切换协议） 请求者已要求服务器切换协议，服务器已确认并准备切换。

**http状态返回代码 2xx （成功）**  
表示成功处理了请求的状态代码。

**http状态返回代码** 代码   说明  
200   （成功）  服务器已成功处理了请求。 通常，这表示服务器提供了请求的网页。  
201   （已创建）  请求成功并且服务器创建了新的资源。  
202   （已接受）  服务器已接受请求，但尚未处理。  
203   （非授权信息）  服务器已成功处理了请求，但返回的信息可能来自另一来源。  
204   （无内容）  服务器成功处理了请求，但没有返回任何内容。  
205   （重置内容） 服务器成功处理了请求，但没有返回任何内容。  
206   （部分内容）  服务器成功处理了部分 GET 请求。

**http状态返回代码 3xx （重定向）**  
表示要完成请求，需要进一步操作。 通常，这些状态代码用来重定向。

**http状态返回代码** 代码   说明  
300   （多种选择）  针对请求，服务器可执行多种操作。 服务器可根据请求者 (user agent) 选择一项操作，或提供操作列表供请求者选择。  
301   （永久移动）  请求的网页已永久移动到新位置。 服务器返回此响应（对 GET 或 HEAD 请求的响应）时，会自动将请求者转到新位置。  
302   （临时移动）  服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。  
303   （查看其他位置） 请求者应当对不同的位置使用单独的 GET 请求来检索响应时，服务器返回此代码。

304   （未修改） 自从上次请求后，请求的网页未修改过。 服务器返回此响应时，不会返回网页内容。  
305   （使用代理） 请求者只能使用代理访问请求的网页。 如果服务器返回此响应，还表示请求者应使用代理。  
307   （临时重定向）  服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。

**http状态返回代码 4xx（请求错误）**  
这些状态代码表示请求可能出错，妨碍了服务器的处理。

**http状态返回代码** 代码   说明  
400   （错误请求） 服务器不理解请求的语法。  
401   （未授权） 请求要求身份验证。 对于需要登录的网页，服务器可能返回此响应。  
403   （禁止） 服务器拒绝请求。  
404   （未找到） 服务器找不到请求的网页。  
405   （方法禁用） 禁用请求中指定的方法。  
406   （不接受） 无法使用请求的内容特性响应请求的网页。  
407   （需要代理授权） 此状态代码与 401（未授权）类似，但指定请求者应当授权使用代理。  
408   （请求超时）  服务器等候请求时发生超时。  
409   （冲突）  服务器在完成请求时发生冲突。 服务器必须在响应中包含有关冲突的信息。  
410   （已删除）  如果请求的资源已永久删除，服务器就会返回此响应。  
411   （需要有效长度） 服务器不接受不含有效内容长度标头字段的请求。  
412   （未满足前提条件） 服务器未满足请求者在请求中设置的其中一个前提条件。  
413   （请求实体过大） 服务器无法处理请求，因为请求实体过大，超出服务器的处理能力。  
414   （请求的 URI 过长） 请求的 URI（通常为网址）过长，服务器无法处理。  
415   （不支持的媒体类型） 请求的格式不受请求页面的支持。  
416   （请求范围不符合要求） 如果页面无法提供请求的范围，则服务器会返回此状态代码。  
417   （未满足期望值） 服务器未满足"期望"请求标头字段的要求。

**http状态返回代码 5xx（服务器错误）**  
这些状态代码表示服务器在尝试处理请求时发生内部错误。 这些错误可能是服务器本身的错误，而不是请求出错。

**http状态返回代码** 代码   说明  
500   （服务器内部错误）  服务器遇到错误，无法完成请求。  
501   （尚未实施） 服务器不具备完成请求的功能。 例如，服务器无法识别请求方法时可能会返回此代码。  
502   （错误网关） 服务器作为网关或代理，从上游服务器收到无效响应。  
503   （服务不可用） 服务器目前无法使用（由于超载或停机维护）。 通常，这只是暂时状态。  
504   （网关超时）  服务器作为网关或代理，但是没有及时从上游服务器收到请求。  
505   （HTTP 版本不受支持） 服务器不支持请求中所用的 HTTP 协议版本。

一些常见的**http状态返回代码**为：

**200 - 服务器成功返回网页  
404 - 请求的网页不存在  
503 - 服务不可用**