

挑战

进制及其运算确立了计算机科学中最基本的数据格式、记录方式；掌握了编程语言，工程师就可以让计算机按自己的想法运行程序了。但做到这些这就够了吗？

为什么有些同学虽然找到了工作，但一直都只能做简单的CRUD呢？为什么大厂面试的时候都喜欢考算法？为什么有些开源的中间件能支撑很大的并发流量，而另外一些功能相同的中间件性能却非常差呢？

说明

大多数小公司的工程师需要处理的数据量很少，只要完成业务功能就可以，学不学数据结构和算法没有任何差别。大厂就完全不同了——大厂的工程师可能会面对几千万甚至几亿的注册用户，开发的是TB、PB级别的数据处理系统，需要利用各种中间件整合衔接多个上下游系统——可用性、健壮性、响应速度这些最基本的性能指标是工程师时时刻刻都要关注和解决的问题。一个看似简单的使用ArrayList还是LinkedList的决定，就可能会造成系统几千倍的性能差别——这既是整个计算机科学最让人着迷的地方，也是最能体现思维模式、开发水平、动手能力的领域。

任务一

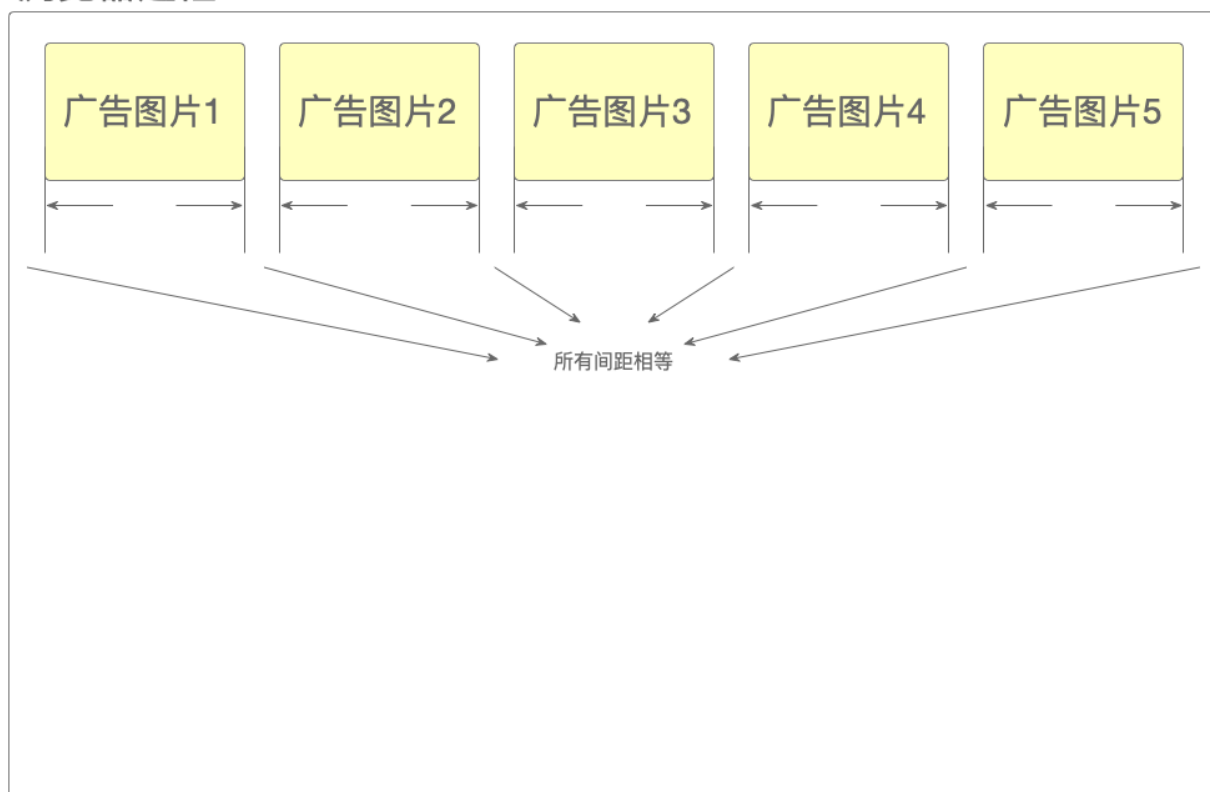
- 首先，能够用HTML + CSS + JavaScript在页面正中生成一幅广告图片

浏览器边框



- 其次，用JavaScript数组在页面上放置多张广告图片，同时动态计算不同广告位之间的布局，实现水平等间距布局

浏览器边框



- 最后，用JavaScript代码实现多张广告图片轮播效果：在页面正中每隔3秒切换不同的广告图片，多张图片轮流显示

浏览器边框



任务二

- 在code.org上以「所见即所得（WYSIWYG）」的编码方式完成「应用实验室」系列任务
 - 变量、条件和函数相关任务（<https://studio.code.org/s/csp4-2020>）
 - 列表、循环和遍历相关任务（<https://studio.code.org/s/csp5-2020>）
 - 计算机程序算法相关的任务（<https://studio.code.org/s/csp6-2020>）
 - 可选任务：参数、返回值和库（<https://studio.code.org/s/csp7-2020>）
- 加深对代码逻辑和程序语法的理解

拓展任务

- 用HTML + CSS + JavaScript实现网页上常见的「省市区多级联动下拉菜单」（理解树型数据结构）
- 选择直辖市的过程



- 选择省份的过程

— 省份 —

北京市

天津市

上海市

重庆市

台湾省

湖北省

— 市 —

— 区 —

湖北省

— 市 —

武汉市

宜昌市

黄冈市

咸宁市

十堰市

神农架林区

黄石市

襄阳市

鄂州市

潜江市

— 区 —

湖北省

武汉市

— 区 —

江汉区

硚口区

武昌区

汉阳区

洪山区

江夏区

江岸区

青山区

东西湖区

汉南区

蔡甸区

黄陂区

新洲区

湖北省

武汉市

江汉区

- 任务提示
 - 不必完全参照上图中的下拉列表，数据可以自己造
 - 淡黄色表示鼠标悬浮状态的背景，可自行设定颜色
- 熟练掌握2~3种常见的计算机算法（例如冒泡排序：<https://algorithm-visualizer.org/brute-force/bubble-sort>）

图形监控器，追踪图形变化

数组监控器，追踪数据变化

日志输出监控器，追踪过程变化

可视化控制器

代码运行过程监视器

这里是 JavaScript，可选择其他语言

- 使用[算法可视化网站](#)运行一种你不熟悉的排序或者查找算法
- 尝试把它用JavaScript代码再写一遍并能成功执行
- 将代码运行后的效果和网站上运行的效果进行对照，看看结果是否一致
- 重复以上步骤，直到能够熟练掌握2~3种算法为止

资源（仅供参考）

- 《编程之美》
- 《编程珠玑》
- 《计算机程序设计艺术 卷1：基本算法(第3版)》
- 《代码大全(第2版)》
- 《大话数据结构》
- 《算法图解》
- 《编程风格：好代码的逻辑》
- 程序控制结构: <https://baike.baidu.com/item/程序控制结构>
- 数据结构与算法: <https://github.com/trekhleb/javascript-algorithms>
- 算法可视化网站
 - <https://algorithm-visualizer.org/>
 - <https://visualgo.net/zh>
 - <https://www.cs.usfca.edu/~galles/visualization/Algorithms.html>

自测

- 实现对数组[0,9,12,1,6,3,7,11]的冒泡排序
- 解释JavaScript中的堆和栈数据结构的区别
- a=1,b=2, 使用至少三种方法交换变量a和b的值
- 使用for循环求出0~300之间的奇数之和
- 去除数组[8, 7, 2, 1, 5, 0, 1, 9, 2]中重复的值, 相同的值只保留一个
- 使用非递归方式对数组[8, 7, 12, 1, 5, 0, 6, 9, 2]执行折半查找