|  |  |
| --- | --- |
| 2017/8/22 上午（8:30-11:30）三个小时三个段 | 产出： |
| 【代码】~9:30计划1：2算法+2sql语句+设计模式（算法，代码练习） | （已经足够，第一道sql题就学习到很多，这就行哦。哈哈）  一篇文章。 |
| 【广度】~10:30一个机器学习算法（前沿知识） | 其他的两分钟都用不上。 |
| 【深度】~11:30整理一个别人的面经（整理）（） |  |

|  |  |
| --- | --- |
| 2017/8/22 下午（2:30-5:30）三个小时三个段 | 产出： |
| 【代码】~15:30计划1：2算法+2sql语句+设计模式（算法，代码练习） | （已经足够，第一道sql题就学习到很多，这就行哦。哈哈）  一篇文章。 |
| 【广度】~16:30一个机器学习算法（前沿知识） |  |
| 【深度】~17:30整理一个别人的面经（整理）（） |  |

|  |  |
| --- | --- |
| 2017/8/22 下午（7:30-10:20）三个小时三个段 | 产出： |
| 【代码】~19:30计划1：2算法+2sql语句+设计模式（算法，代码练习） | （已经足够，第一道sql题就学习到很多，这就行哦。哈哈）  一篇文章。 |
| 【广度】~20:30一个机器学习算法（前沿知识） | 一：看以前的maven构建环境。当时做烟基的时候，那是因 |
| 【深度】~21:30整理一个别人的面经（整理）（） |  |
| 【总结】~22:20整理一个别人的面经（整理）（） |  |

每天必须要做的事情：代码，复习，日记，算法，别人的面经（这个无论是你自己现在还是以后面试官。还是你现在应该要补充的知识点，都非常重要。）

|  |  |
| --- | --- |
| 2017/8/22 上午（8:30-11:30）三个小时三个段 | 产出： |
| 【代码】~9:30计划1： | 到哥哥哪儿去了，所以上午没有及时到办公室。  不过可以改进的是，自己晚上入睡的时间，到床上就不要玩手机了，免得打扰到别人休息。这样对你自己也是一种好的监督，让自己早点睡。挺好的。 |
| 【广度】~10:30一个机器学习算法（前沿知识） | 1.看了一小节的algorithm。其实不单单是学习算法，有时候学习英语也行啊，这样挺好的，提升各个方面的能力。 |
| 【深度】~11:30整理一个别人的面经（整理）（） | 2.实验室的电脑机器坏了，拆开修了一下。警报一下就找到了突破口，挺好的，换了一根4g的内存条，挺好的。  3，台式机，从给家里买电脑那次应该整个台式机都能摸熟练了，所以说还是实践进步快。  4.985给你的，不仅仅是文凭。而应该是一种敢为人先，一种勇气，一种追求卓越的心。当你的同学都不敢入门机器学习的时候，你在马老师和董保华的建议下，已经有了志愿和学习基础了。这就是追求，牛逼的地方。  4．在应届生上投出了几份大数据开发相关的简历。不过都对数据挖掘算法，或者统计学，数据分析有些要求，所以要开始学习了。你相当于是第一批人哦。加油。 |

|  |  |
| --- | --- |
| 2017/8/22 下午（2:30-5:30）三个小时三个段 | 产出： |
| 【代码】~15:30计划1：2算法+2sql语句+设计模式（算法，代码练习） | 下午看了4节machine learning algorithm。相当于复习英语，复习数学，perfect。 |
| 【广度】~16:30一个机器学习算法（前沿知识） | Also learning course of Machine learning。挺好 |
| 【深度】~17:30整理一个别人的面经（整理）（） | 一道华为的编程题，从思路—做的过程中优化改进，强化了之前做题过程中遇到的一些问题。实践中回顾。非常好。  Linked 1 见附件1 |

|  |  |
| --- | --- |
| 2017/8/22 下午（7:30-10:20）三个小时三个段 | 产出： |
| 【代码】~19:30计划1：2算法+2sql语句+设计模式（算法，代码练习） | （已经足够，第一道sql题就学习到很多，这就行哦。哈哈）  一篇文章。 |
| 【广度】~20:30一个机器学习算法（前沿知识） | 一：看以前的maven构建环境。当时做烟基的时候，那是因 |
| 【深度】~21:30整理一个别人的面经（整理）（） |  |
| 【总结】~22:20整理一个别人的面经（整理）（） |  |

|  |  |
| --- | --- |
| 华为机试 |  |
| 连续输入字符串，请按长度为8拆分每个字符串后输出到新的字符串数组；  •长度不是8整数倍的字符串请在后面补数字0，空字符串不处理。  **输入描述:**  连续输入字符串(输入2次,每个字符串长度小于100)  **输出描述:**  输出到长度为8的新字符串数组  **示例1**  **输入**  abc  123456789  **输出**  abc00000  12345678  90000000 | |
| 思路：（最终的）   1. 先把数据全部打印出来。【充分利用oj的特性，最后结果是打印出来的东西和设定的结果进行文件比对，这个相对来说比leetcode来说要简单很多。】   2.再统一计算不足的部分8 的倍数的部分。打印输出即可，非常nice。这样就可以解决问题了。 | |
| **public** **class** DistributeString {  **public** **static** **void** main(String[] args) {  // 用一个函数处理。  Scanner scanner = **new** Scanner(System.***in***);  String str1 = scanner.next();  String str2 = scanner.next();  *change*(str1);  *change*(str2);  }  **public** **static** **void** change(String str) {  // 计算长度，看需要几次输出。  **if** (str == **null** || str.equals("")) **return** ;【2】  **int** num = str.length();  **char**[] c =str.toCharArray();  // int j = 0;  // for (int i = num,j = 0; i > 0【3】;i--,j++) {  // System.out.print(c[j]);  // if ((j + 1) % 8 == 0) {  // System.out.println();  // }  // }  **for** (**int** i = 0; i < c.length; i++) {  System.***out***.print(c[i]);  **if** ((i + 1) % 8 == 0) {  System.***out***.println();  }  }  // 计算需要添加的0的个数  **int** addnum = 0;  **if**(num%8 != 0)  addnum = 8 - num % 8;  **for** (**int** i = 0; i < addnum ; i++) {  System.***out***.print(0);  }  System.***out***.println();【1】  }  } | **public** **class** DistributeString {  **public** **static** **void** main(String[] args) {  // 用一个函数处理。  Scanner scanner = **new** Scanner(System.***in***);  String str1 = scanner.next();  String str2 = scanner.next();  *change*(str1);  System.***out***.println();【1】  *change*(str2);  }  **public** **static** **void** change(String str) {  // 计算长度，看需要几次输出。  **if** (str == **null** || str.equals("")) **return** ;  **int** num = str.length();  **char**[] c =str.toCharArray();  // int j = 0;  // for (int i = num,j = 0; i > 0;i--,j++) {  // System.out.print(c[j]);  // if ((j + 1) % 8 == 0) {  // System.out.println();  // }  // }  **for** (**int** i = 0; i < c.length; i++) {  System.***out***.print(c[i]);  **if** ((i + 1) % 8 == 0) {  System.***out***.println();  }  }  // 计算需要添加的0的个数  **int** addnum = 0;  **if**(num%8 != 0)  addnum = 8 - num % 8;  **for** (**int** i = 0; i < addnum ; i++) {  System.***out***.print(0);  }  }  } |
| 【1】这种只有oj做多了才能立马可以改正出来 | |
| 【2】虽然返回值是void，但是还是可以用return来控制我们的程序。 | |
| 【3】有时候这样写了，这点错误很难被发现，是满足这个条件才会执行这个循环体。 | |
| 【反思】在nowcoder上，很多oj上就是文件比对，那么应该优先想到的是输出正确结果就行。这个没有做好。而不是最开始想着改变string的内容，不是真的在string的尾部加上0.其实如果加上0 也是可以的，还是直接计算缺多少个0即可。 | |
| 別人的解法：  import java.util.\*;  public class Main{  public static void main(String[] args){  Scanner sc = new Scanner(System.in);  while(sc.hasNext()){  String s = new String(sc.nextLine());  if(s.length()%8 !=0 )  s = s + "00000000";  while(s.length()>=8){  System.out.println(s.substring(0, 8));  s = s.substring(8);  }  }  }  }  很好的思路。把长度不能整除8整个字符串都加上8个0.然后再输出，输出长度为8的子串。直到str的长度低于8时退出循环。  这个思路非常好。  之前在leetcode里也有一个题目大概也是这个意思。不合格的条件先让他加上一倍，相当于把数据进行了标记。如果是正负数好像都可以这样思考一下。 | |
| 【学习】string的一些常用方法。比如substring();方法。  **"unhappy".substring(2) returns "happy"**  **"Harbison".substring(3) returns "bison"**  **"emptiness".substring(9) returns "" (an empty string)**  "hamburger".substring(4, 8) returns "urge"  "smiles".substring(1, 5) returns "mile"  http://tool.oschina.net/apidocs/apidoc?api=jdk-zh | |
| 提高：用python写一遍。 | |
|  | |
|  | |

简历：

|  |
| --- |
| 潜入IT圈五年之久、看过数万份简历的HR，教你如何写简历。HR在筛选简历时主要从公司需求出发，重点不一，不过还是有很多“通用”的套路，为了在30秒内判断出这份简历是否值得跟进，我认为程序员写简历的正确姿势是这样的：  **一、基本格调**  即打开简历之后的第一印象。就好比我们看见一个人，会有一个整体的感觉，他是fashion的、小清新的还是老道的？有了第一印象之后再慢慢分解来看。  **加分写法：**   * 简洁明了，逻辑结构清晰。 * 字体，排版，顺畅，清晰整齐就好。 * 最好是PDF格式，兼容性强且不易乱序。   **减分写法：**   * 设计的过于浮夸或者过于简单的。（eg.有的简历五颜六色、非常酷炫，却半天找不到联系方式，抑或是只有个人基本信息和公司名称) * 写了十几页，半天打不开的，或者加载了半天，打开还乱码。   **二、基本信息（姓名/性别/毕业院校/电话/邮箱/居住地/期望地）**  **加分写法:**   * 清晰罗列出以上信息，这样HR就不用在接下来的电话沟通或面试中再去追问这些内容，建立我们接下来电话沟通对你的熟悉度。 * 再额外能加上QQ或者微信就更好了（以防有时候电话打不通哦，时不时会遇到这种情况）   **减分写法：**   * 大部分的基本信息没有写 * 甩给我一个Github链接，极致简洁的几句描述，需要通过你的链接来找你的联系方式。（如果不是博客写的特别好，基本是要放弃你了）   **三、工作经历&项目经历**  **加分写法：**   * 工作经历项目经历可参照万能的STAR法则来写，STAR不清楚的童鞋点[这里](https://link.zhihu.com/?target=http%3A/www.xuexila.com/success/story/509956.html" \t "_blank)啦 * 效力过哪些公司，我们匹配的公司？ BAT？ 知名大型互联网公司？ * 做过什么行业领域，和我们目前的行业是否匹配 * 擅长的技术语言，应用了哪些技术栈，（Java, Scala，Ruby, React, Vue, Microservice…） * 经历的项目复杂度，及在项目中承担什么样的角色(人的变化/技术的变化/环境的变化/不同工作经历相同角色的不同点) * 时间节点（空档期）   **减分写法：**   * 看了半天，不知所云，没有任何亮点，没有让人有去和你聊一聊深扒的信息。   **来几个栗子**  栗子1错误打开方式：   * XX（全栈工程师）2013.06 — 至今 * 参与需求分析及实现方案设计。 * 设计数据库表结构，实现后台功能及web页面展示。 * 产品线上部署及运维。 * ay 配置管理工程师 2010.03 — 2013.03 * 负责公司产品性能测试，及线上数据分析 * 负责公司配置管理，环境维护等工作   **点评：看不出来他做的什么事情，没有逻辑性，甚至不知道他做的什么技术语言。**  **栗子2正确打开方式：**  **西安XXX公司 Java工程师 — 2016.2月-2017.2月**  1、MOGU推荐架构数据与缓存层设计开发   * MOGU是一款时尚资讯app,负责推荐页面资讯feed流的展示及用户历史的展示 * 负责数据层,处理前端逻辑整个开发工作,分布式rpc服务搭建 * 负责进行压测监测、缓存处理,对接又进行改进优化,主用redis缓存   2、基于JAVA的电商爬虫开发   * 使用java搭建爬虫server平台,进行配置和开发,进行网页改版监测功能开发 * 爬取淘宝时尚品牌与其他电商网站商品品牌与详情等 * 通过频率、ip池、匿名代理等应对一些网站的反爬   3、同图搜索Solr服务开发  基于算法组的同图策略,使用solr做java接又实现rpc服务搭建,进行索引构建和solr实现  **北京XXX**  java大数据工程师— 2013.4月-2015.12月  1、负责实时流消息处理应用系统构建和实现   * 在调研了kafka的优势和我们的具体需求之后,用kafka作为消费者,保证高吞吐处理消息,并持久化消息的同时供其它服务使用,进行了系统的设计和搭建使用。 本地日志保证消息不丢失,并通过记录游标滑动重复读取数据。 * 使用storm 负责搭建消息处理架构,并完成基于业务的消息落地,提供后续的数据 统计分析实时和离线任务,诸如pv、uv等数据,为运营做决策 * 网站用户行为埋点和基于js的日志收集器开发,定义接又和前端部门配合。主用go 2、hadoop集群搭建和数据分析处理   2、基于CDH的集群搭建工作,后期进行维护  编写MapReduce程序,能将复杂工作逻辑化,尽最大能力发挥大数据应用的特点, 对程序高要求,监控自己程序运行情况,使用内存合理,注重增量和全量运算的利弊  3、调度系统设计与实现 基于quartz2搭建调度平台,带徒弟实现相关功能并定期review代码  4、数据库调优 负责主从搭建,并掌握主从搭建的利弊,了解业界mycat原理,有数据库优化经验,能 正确并擅长使用索引,对锁有深刻的认识  5、网站开发 java web网站业务开发,并能很好的使用缓存技术,对重构有实际的经验,并对面向对 象开发有全面的实战经验。了解java数据结构的使用场景,虽然对于大并发没有太大的 发挥余地,但是掌握了数据结构,对于并发和阻塞等有自己的见解。  **点评：非常清晰的告诉简历阅读者自己做了什么事情，负责了什么样的事情，用了什么技术栈，且逻辑连贯。**  **四、 工作期望&个人评价**  加分写法：   * 对自己有一个全方位的一个描述总结，让别人更好的解读你。或者在此处，高亮你的优点特长有哪些。 * 即使不写个人评价，也一定记得写上工作期望。   减分写法：  完全看不出个性特点，写和没写没什么区别。 来几个栗子  **栗子1 错误打开方式**  为人性格,诚实谦虚，勤奋，能吃苦耐劳，有耐心，有团队意识，能和同学和谐相处，能虚心接受别人的建议的人。  责任心强，善于沟通，具有良好的团队合作精神；专业扎实，具有较强的钻研精神和学习能力；性格比较乐观外向，喜欢打羽毛球。  **栗子2正确打开方式**   * 我对自己的定位: 主攻前端,同时在其他方面打打辅助。我不希望过于依赖别人,即使没有后端没有设计没有产品经理,我依然想要把这个产品做到完美。毕竟全栈才能最高效地解决问题。 * 我对工作的态度: 第一,要高效完成自己的本职工作。第二,要在完成的基础上寻找完美。第三,要在完美的基础上,与其他同事 互相交流学习,互相提升。工作是一种生活方式,不是一份养家糊口的差事。 * 我怎样克服困难: 不用百度是第一原则,在遇到技术问题时我往往会去Google、Stack over flow上寻找答案。但通常很多问题 并不一定已经被人解决,所以熟练地阅读源码、在手册、规范甚至 REPL的环境自己做实验才是最终解决问题的办法。相信事实的结果,自己动手去做。 * 怎样保持自己的视野:我一直认为软件开发中视野极其重要,除了在 Twitter 上关注业界大牛,Github Trending 也是每周必刷。 另外 Podcast、Hacker News、Reddit 以及TechRadar 也是重要的一手资料。保持开阔视野才能找到更酷的解决方案。 * 我的优势: 热爱技术、自学能力强,有良好的自我认知。全面的技能树与开阔的视野,良好的心态、情商与沟通能力。 * 我的劣势: 非科班出身没有科班同学对算法的熟练掌握,但我决定死磕技术,弥补不足。   **栗子3正确打开方式**   * 极客、热爱技术、热爱开源 * Ruby on Rails：精通 * Agile/Lean：精通 * ReactJS：掌握 * Docker：掌握 * AWS：掌握   **五、 是否有博客，个人技术栈点等**   1. 看到有这项的HR两眼已经放光了，加分加分项，说明你真正的热爱技术，善于学习总结，乐于分享，且有投入自己的业余时间到软件事业中。 2. 我喜欢的书籍:《重构》《卓有成效的程序员》《代码整洁之道》等 3. 我喜欢的社区: 图灵社区，知乎，博客园，Stack Over flow，Google Developer Group等 4. 我的博客链接、个人作品链接如下:  * [https://github.com/github](https://link.zhihu.com/?target=https%3A/github.com/github) * [http://www.oschina.net/](https://link.zhihu.com/?target=http%3A/www.oschina.net/) * [https://www.cnblogs.com/](https://link.zhihu.com/?target=https%3A/www.cnblogs.com/) * [https://itunes.apple.com/app/battle-of-crab/id1121917063?l=en&amp;amp;amp;mt=8](https://link.zhihu.com/?target=https%3A/itunes.apple.com/app/battle-of-crab/id1121917063%3Fl%3Den%26mt%3D8)   **六、简历内容真实性**  老司机提醒你，你简历的任意一个细节将会是后面面试中的呈堂证供。  基本就这些了，希望对大家能有帮助，看起简历来几十秒，码字还是个体力活。 |

面经

|  |
| --- |
| 唯品会 南京大学。  1 hashmap以及1.8hashmap新特点  2 springmvc 处理请求过程  3 反射机制  4 反射机制在springmvc中的应用  5 1000个数找出最大的一个？找出最大10个  6 5亿个URL，找出一个  7 二叉树在数据库中存储 设计表  8 TCP UDP 以及三次握手 时常大概25分钟 |
| :给一个二维平面，而且横纵坐标都不会重复（简化了排序），要求“不存在左上方还有点”的点集。因为数据量最大是50W，所以基本上用是O(nlogn)的方法解决。  首先按x坐标排个序（因为y不重复所以不用管），然后从后往前（此时保证当前点的x是比后面的x要小的），记录一个当前最大的Y，如果当前这个位置的y比Y还要大，那么明显这个位置的“左上方”不可能有点了。问题解决。 |
| 美团成都一面总结  今天晚上面了美团成都，整体还是比较基础，属于基础面吧。总结一下(求赞。。。。)  1、 HTTP请求的过程，越清楚越好（我从dns解析到TCP三次握手，四次挥手，还有后台请求逻辑讲了一遍）  2、 Java线程池ThreadPoolEcecutor参数，基本参数，使用场景  3、 HashMap、HashTable、CurrentHashMap的区别及场景，说一下你的理解。HashMap的负载因子，干啥用的。  4、 ArrayList与LinkedList的区别及使用场景，有没有ArrayList增删效率比LinkedList高的场景。ArrayList底层采用什么实现数据的拷贝。  5、 MySQL的ACID讲一下，延伸到隔离级别  6、 JMM讲一下，volite关键字及使用场景。 （内存模型）  7、 JVM的内存布局以及垃圾回收原理及过程讲一下。讲一下CMS垃圾收集器垃圾回收的流程。以及CMS的缺点（忘了。。）  8、 智力题（聊着聊着突然来个），两个杯子，一个有3升水，一个有5升水，请问如何倒出4升水。。（我回答说，使两者水平面一致。。。。一看到题，就这样啊。。但其答案不是我回答的这。。）  9、 手写二叉树的中序遍历与层序遍历  10、 手写二分查找（很基础）  11、最后还聊了一下项目，聊一下难点。  12、最后就是你有什么要问的。1、表现2、你们那边主要用那些java技术3、如果过，啥时通知下一面结果。  祝大家好运！整体还是比较基础的。。（58分钟） |
|  |