

MVC 相关

MVC 工作原理

1. model 数据模型操作层，是应用程序中用于处理应用程序数据逻辑的部分
2. view 视图层，是应用程序中处理数据显示的部分。
3. controller 业务处理层，是应用程序中处理用户交互的部分。

单一入口

工作原理

用一个处理程序文件处理所有的 HTTP 请求 根据请求时的参数的不同区分不同的模块和操作请求

优势

1. 可以进行统一的安全性检查
2. 集中处理程序

劣势

1. URL 不美观(解决方法：URL 重写)
2. 处理效率会降低(可忽略)

模板引擎

PHP 是一种 HTML 内嵌式在服务端执行的脚本语言，但是 PHP 又很多可以使 PHP 代码和 HTML 代码分开的模板引擎，例如：smarty

工作原理

模板引擎就是庞大的完善的正则表达式替换库

算法

排序算法

冒泡排序

原理：两两相邻的数进行比较，如果反序就交换，否则不交换

时间复杂度：最坏($O(n^2)$)，平均($O(n^2)$)

空间复杂度： $O(1)$

快速排序

原理：通过一趟排序将要排序的数据分割成独立的两部分，其中一部分的所有数据和另一部分的所有数据都要比它小，然后按照此方法对这两部分数据分别进行快速排序，整个排序过程可以递归完成

时间复杂度：最坏($O(n^2)$)，平均($O(n \log_2 n)$)

空间复杂度：最差($O(n)$)，平均($O(\log_2 n)$)

直接插入排序

原理：每次从无序表中取出第一个元素，把他插入到有序表的合适位置，使有序表仍然有序

时间复杂度：最坏($O(n^2)$), 平均($O(n^2)$)

空间复杂度： $O(1)$

选择排序

原理：每次从待排序的数据元素中选出最小(或最大)的一个元素，存放在序列的起始位置，知道全部待排序的数据元素排外

时间复杂度：最坏($O(n^2)$), 平均($O(n^2)$)

空间复杂度： $O(1)$

希尔排序

原理：把待排序的数据根据增量分成几个子序列，对子序列进行插入排序，知道增量为 1，直接插入进行插入排序；增量的排序，一般是数组的长度的一半，再变为原来增量的一半，直到增量为 1

时间复杂度：最坏($O(n^2)$), 平均($O(n\log_2 n)$)

空间复杂度： $O(1)$

堆排序

原理：把待排序的元素按照大小在二叉树位置上排序，排序好的元素要满足：父节点的元素要大于子节点；这个过程叫做堆化过程，如果根节点存放的最大的数，则叫做大根堆，如果是最小，就叫小跟堆，可以把根节点拿出来，然后再堆化，循环到最后一个节点

时间复杂度：最坏($O(n\log_2 n)$), 平均($O(n\log_2 n)$)

空间复杂度： $O(1)$

归并排序

原理：将两个（或两个以上）有序表合并成一个新的有序表，即把待排序序列分为若干个有序的子序列，再把有序的子序列合并为整体有序序列

时间复杂度：最坏($O(n\log_2 n)$), 平均($O(n\log_2 n)$)

空间复杂度： $O(n)$

查找算法

二分查找

原理：从数组的中间元素开始，如果中间元素正好是要查找的元素，搜索结果，如果某一个特定元素大于或者小于中间元素的那一半中查找，而且跟开始一样从中间开始比较，如果某一步骤数组为空，代表找不到

时间复杂度：最坏($O(n\log_2 n)$), 平均($O(n\log_2 n)$)

空间复杂度：迭代($O(1)$), 递归($O(\log 2n)$)

顺序查找

原理：按一定的顺序检查数组中每一个元素，直到要找到所要寻找的特定指为止

时间复杂度：最坏($O(n)$), 平均($O(n)$)

空间复杂度： $O(1)$



微信二维码扫一扫咨询大神进阶课程内容

微信号码 若兰：**zqf19907493857**

QQ 咨询联系 妮妮：**194210485**