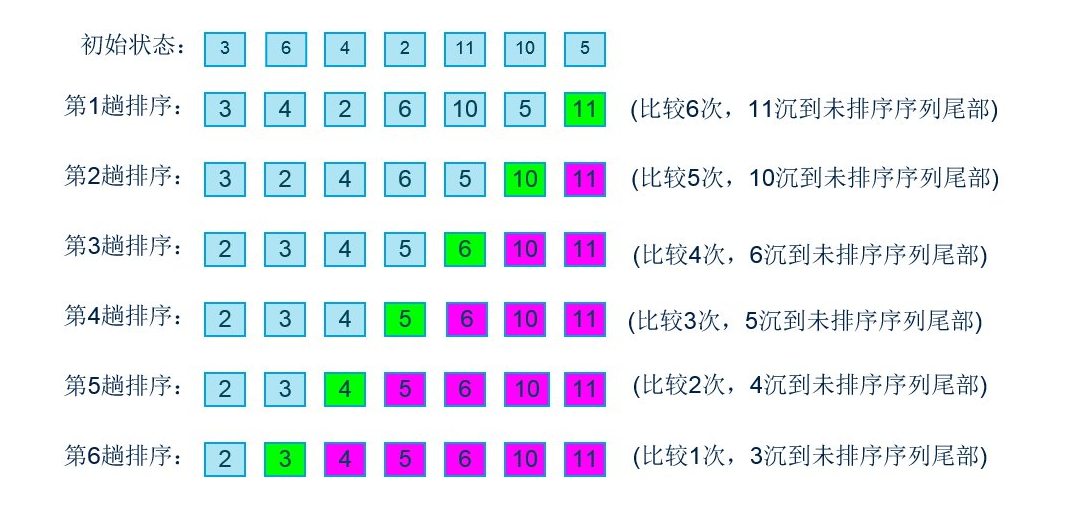
冒泡排序算法的原理如下：

1. 比较相邻的元素。如果第一个比第二个大，就交换他们两个。
2. 对每一对相邻元素做同样的工作，从开始第一对到结尾的最后一对。在这一点，最后的元素应该会是最大的数。
3. 针对所有的元素重复以上的步骤，除了最后一个。
4. 持续每次对越来越少的元素重复上面的步骤，直到没有任何一对数字需要比较。



时间复杂度

若文件的初始状态是正序的，一趟扫描即可完成排序。所需的关键字比较次数

https://gss0.bdstatic.com/-4o3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D12/sign=5e6ca6622c738bd4c021b633a08bb91b/b3b7d0a20cf431ad4f379a864936acaf2fdd98ac.jpg

和记录移动次数

https://gss1.bdstatic.com/-vo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D16/sign=c765e07cd5ca7bcb797bc329bf09e8ac/48540923dd54564e27333d86b1de9c82d1584f3f.jpg

均达到最小值：

https://gss1.bdstatic.com/-vo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D84/sign=2cfb4bf49182d158bf8254b5810a024c/8326cffc1e178a8244f3b2bafb03738da877e8da.jpg

，

https://gss3.bdstatic.com/7Po3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D66/sign=1745c99097cad1c8d4bbff217e3e11cd/b21c8701a18b87d6007472a0050828381f30fd5f.jpg

。

所以，冒泡排序最好的[时间复杂度](https://baike.baidu.com/item/%E6%97%B6%E9%97%B4%E5%A4%8D%E6%9D%82%E5%BA%A6)为

https://gss3.bdstatic.com/7Po3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D33/sign=feddd1e257fbb2fb302b5e114e4a94f7/b58f8c5494eef01f9d1ac17ae2fe9925bc317d07.jpg

。  
　　若初始文件是反序的，需要进行

https://gss3.bdstatic.com/7Po3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D32/sign=411c7ed7d309b3deefbfe26acdbf0492/8b82b9014a90f60364df45f83b12b31bb051ed26.jpg

趟排序。每趟排序要进行

https://gss3.bdstatic.com/7Po3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D30/sign=3af139940cf41bd5de53eef450da752f/503d269759ee3d6d1214597441166d224e4adebb.jpg

次关键字的比较(1≤i≤n-1)，且每次比较都必须移动记录三次来达到交换记录位置。在这种情况下，比较和移动次数均达到最大值：

https://gss1.bdstatic.com/-vo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D171/sign=f38fc3a1bb389b503cffe455b434e5f1/838ba61ea8d3fd1f846a7b85324e251f95ca5f2a.jpg

https://gss2.bdstatic.com/9fo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D188/sign=752dcc96713e6709ba0041f703c69fb8/34fae6cd7b899e51dd180aa54ca7d933c9950de8.jpg

冒泡排序的最坏时间复杂度为

https://gss1.bdstatic.com/-vo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D40/sign=17b28fa860d0f703e2b294dc09fa9d5c/e4dde71190ef76c63a4e22949f16fdfaae5167f5.jpg

。[1]

综上，因此冒泡排序总的平均时间复杂度为

https://gss1.bdstatic.com/-vo3dSag_xI4khGkpoWK1HF6hhy/baike/s%3D40/sign=17b28fa860d0f703e2b294dc09fa9d5c/e4dde71190ef76c63a4e22949f16fdfaae5167f5.jpg

。

算法稳定性

冒泡排序就是把小的元素往前调或者把大的元素往后调。比较是相邻的两个元素比较，交换也发生在这两个元素之间。所以，如果两个元素相等，是不会再交换的；如果两个相等的元素没有相邻，那么即使通过前面的两两交换把两个相邻起来，这时候也不会交换，所以相同元素的前后顺序并没有改变，所以冒泡排序是一种稳定排序算法。