

vector与arraylist.txt比较Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/629/2021\\_2022\\_vector\\_E4\\_B8\\_8Ea\\_c104\\_629787.htm](https://www.100test.com/kao_ti2020/629/2021_2022_vector_E4_B8_8Ea_c104_629787.htm)

Vector 还是ArrayList——哪一个更好，为什么？要回答这个问题不能一概而论，有时候使用Vector比较好.有时是ArrayList，有时候这两个都不是最好的选择。你别指望能够获得一个简单肯定答案，因为这要看你用它们干什么。下面有4个要考虑的因素：API 同步处理 数据增长性 使用模式 下面针对这4个方面进行一一探讨 API 在由Ken Arnold等编著的《Java Programming Language》(Addison-Wesley, June 2000)一书中有这样的描述，Vector类似于ArrayList.。所有从API的角度来看这两个类非常相似。但他们之间也还是有一些主要的区别的。同步性 Vector 是同步的。这个类中的一些方法保证了Vector中的对象是线程安全的。而ArrayList则是异步的，因此ArrayList中的对象并不是线程安全的。因为同步的要求会影响执行的效率，所以如果你不需要线程安全的集合那么使用ArrayList是一个很好的选择，这样可以避免由于同步带来的不必要的性能开销。 数据增长从内部实现机制来讲ArrayList和Vector都是使用数组(Array)来控制集合中的对象。当你向这两种类型中增加元素的时候，如果元素的数目超出了内部数组目前的长度它们都需要扩展内部数组的长度，Vector缺省情况下自动增长原来一倍的数组长度，ArrayList是原来的 50%,所以最后你获得的这个集合所占的空间总是比你实际需要的要大。所以如果你要在集合中保存大量的数据那么使用 Vector有一些优势，因为你可以通过设置集合的初始化大小来避免不必要的资源开销。使用

模式在ArrayList和Vector中，从一个指定的位置(通过索引)查找数据或是在集合的末尾增加、移除一个元素所花费的时间是一样的，这个时间我们用 $O(1)$ 表示。但是，如果在集合的其他位置增加或移除元素那么花费的时间会呈线性增长

： $O(n-i)$ ，其中 $n$ 代表集合中元素的个数， $i$ 代表元素增加或移除元素的索引位置。为什么会这样呢?以为在进行上述操作的时候集合中第 $i$ 和第 $i$ 个元素之后的所有元素都要执行位移的操作。这一切意味着什么呢?这意味着，你只是查找特定位置的元素或只在集合的末端增加、移除元素，那么使用Vector或ArrayList都可以。如果是其他操作，你最好选择其他的集合操作类。比如，LinkedList集合类在增加或移除集合中任何位置的元素所花费的时间都是一样的? $O(1)$ ，但它在索引一个元素的使用缺比较慢- $O(i)$ ，其中 $i$ 是索引的位置.使用ArrayList也很容易，因为你可以简单的使用索引来代替创建iterator对象的操作。LinkedList也会为每个插入的元素创建对象，所有你要明白它也会带来额外的开销。最后，在《PracticalJava》一书中Peter Hagggar建议使用一个简单的数组(Array)来代替Vector或ArrayList。尤其是对于执行效率要求高的程序更应如此。因为使用数组(Array)避免了同步、额外的方法调用和不必要的重新分配空间的操作。更多优质资料尽在百考试题论坛 百考试题在线题库 java认证更多详细资料 100Test 下载频道开通，各类考试题目直接下载。详细请访问 [www.100test.com](http://www.100test.com)