

Performance test 1

Comparison of the runtime between the two static methods *retainAll(Collection collection, Collection toRetain)* and *intersection(List list1, List list2)* of the class *org.apache.commons-collections.ListUtils*. Both essentially compute the intersection of two given parameters. Hence their runtime should be similar.

The method *intersestion(...)* is optimized already.

Task

Find the performance bug in method *retainAll* and try to understand it.

Problem is

Method *retainAll(...)* calls *contains(...)* on the second parameter within a loop. This is dynamically *LinkedList.contains(...)* which calls *LinkedList.indexOf(...)*. This does a linear search on the list. Thus *retainAll(...)* has an asymptotic runtime of $\mathcal{O}(n^2)$.

Solution

Encapsulate the second parameter into a *HashSet* so that the method *contains(...)* runs in $\mathcal{O}(1)$.

Hints

1. The runtime is not consumed by *retainAll(...)* itself.
2. The method *Collection.contains(...)* is actually *LinkedList.contains(...)* dynamically.
3. A lookup in $\mathcal{O}(1)$ can be done with the help of a *HashSet*.