

# Forum:[Hack Forums](#)

Thread: [Arrays vs Collections?](#)

**Folks** - August 3, 2016, 12:01 am

I used to use arrays all the time but now I just stumbled upon collections.

Is there any reason to use arrays as collections seem superior in almost every way and I've read some articles on stack overflow which seem to agree with me.

**csrk** - August 3, 2016, 12:12 am

(Aug 3, 2016 12:01 AM)Folks Wrote: [\[CITE\]](#) Is there any reason to use arrays as collections seem superior in almost every way and I've read some articles on stack overflow which seem to agree with me.

Depends on what you're using it for and what collection it is. For example ArrayList uses delegation on an array, while LinkedList uses linked nodes.

Getting the Nth element from an array is possible with  $O(1)$  complexity, but getting the Nth element from a LinkedList is done in  $O(n)$  complexity (if it's not circularly linked, in which case it is still worse than  $O(1)$ ), as you have to traverse the link in order to find your element before you can return it.

I suggest you take a course in datastructures if your school offers it, or read up on it online. Once you understand how the different structures work and their pros and cons it's easier to choose which one you need.

**Yuikoro** - August 3, 2016, 4:09 am

Following more on what Rewbert said, there's a great site to find out the best kind of datastructure to use for your purpose:

<http://bigocheatsheet.com/>

I don't really use it anymore but it's helpful. It also has stuff for sorting algorithms and more

**domien** - August 3, 2016, 6:40 am

Remember this first rule: Use arrays when you know the length beforehand, otherwise use collections.

There are exceptions of course, but I will not mention them here.

Arraylists have an amortized complexity of  $O(1)$ , though.

**GammaG** - August 3, 2016, 6:44 am

Sometimes you will still have to use array - like for a `String.split("[ ]+");` method

**BeNT-** - August 3, 2016, 6:48 am

All the above are good answers.

For the general rule, use existing optimal code/classes when you can, but know your data set is hugely important. Don't reinvent code unless you really need to. Arrays are extremely useful at low level when you know a set data length in advance. Otherwise big O kicks in as previous posts implied and you are better off using optimized code (unless you really need something specific)

Cheers