

# CSE 143 Midterm Cheat Sheet

## Methods Found in ALL collections (Lists, Stacks, Queues, Sets, Maps)

<code>clear()</code>	removes all elements of the collection
<code>equals(collection)</code>	returns <code>true</code> if the given other collection contains the same elements
<code>isEmpty()</code>	returns <code>true</code> if the collection has no elements
<code>size()</code>	returns the number of elements in the collection
<code>toString()</code>	returns a string representation such as "[10, -2, 43]"

## Methods Found in both Lists and Sets (ArrayList, LinkedList, HashSet, TreeSet)

<code>add(value)</code>	adds value to collection (appends at end of list)
<code>contains(value)</code>	returns <code>true</code> if the given value is found somewhere in this collection
<code>iterator()</code>	returns an <code>Iterator</code> object to traverse the collection's elements
<code>remove(value)</code>	finds and removes the given value from this collection
<code>removeAll(collection)</code>	removes any elements found in the given collection from this one
<code>retainAll(collection)</code>	removes any elements <i>not</i> found in the given collection from this one

## List<E> Methods (10.1)

<code>add(index, value)</code>	inserts given value at given index, shifting subsequent values right
<code>indexOf(value)</code>	returns first index where given value is found in list (-1 if not found)
<code>get(index)</code>	returns the value at given index
<code>remove(index)</code>	removes/returns value at given index, shifting subsequent values left
<code>set(index, value)</code>	replaces value at given index with given value

## Stack<E> Methods

<code>pop()</code>	removes the top value from the stack and returns it; throws an <code>EmptyStackException</code> if the stack is empty
<code>push(value)</code>	places the given value on top of the stack

## Queue<E> Methods

<code>add(value)</code>	places the given value at the back of the queue
<code>remove()</code>	removes the value from the front of the queue and returns it; throws a <code>NoSuchElementException</code> if the queue is empty

For problems involving stacks or queues, you ARE NOT ALLOWED to use for-each loops, iterators, or any operation other than those specified here for stacks/queues.

Queues should be constructed using the `Queue<E>` interface and the `LinkedList<E>` implementation. Stacks should be constructed using the `Stack<E>` class (there is no interface). For example, to construct a queue and a stack of `String` values, you would say:

```
Queue<String> q = new LinkedList<String>();
Stack<String> s = new Stack<String>();
```

To transfer from a queue to a stack:

```
while (!q.isEmpty()) {
    s.push(q.remove());
}
```

To transfer from a stack to a queue:

```
while (!s.isEmpty()) {
    q.add(s.pop());
}
```

# CSE 143 Midterm Cheat Sheet

## Map<K, V> Methods (11.3)

containsKey ( <b>key</b> )	true if the map contains a mapping for the given key
get ( <b>key</b> )	the value mapped to the given key (null if none)
keySet ()	returns a Set of all keys in the map
put ( <b>key, value</b> )	adds a mapping from the given key to the given value
putAll ( <b>map</b> )	adds all key/value pairs from the given map to this map
remove ( <b>key</b> )	removes any existing mapping for the given key
toString ()	returns a string such as "{a=90, d=60, c=70}"
values ()	returns a Collection of all values in the map

## String Methods (3.3, 4.4)

charAt ( <b>i</b> )	the character in this String at a given index
contains ( <b>str</b> )	true if this String contains the other's characters inside it
endsWith ( <b>str</b> )	true if this String ends with the other's characters
equals ( <b>str</b> )	true if this String is the same as <i>str</i>
equalsIgnoreCase ( <b>str</b> )	true if this String is the same as <i>str</i> , ignoring capitalization
indexOf ( <b>str</b> )	first index in this String where given String begins (-1 if not found)
lastIndexOf ( <b>str</b> )	last index in this String where given String begins (-1 if not found)
length ()	number of characters in this String
startsWith ( <b>str</b> )	true if this String begins with the other's characters
substring ( <b>i, j</b> )	characters in this String from index <i>i</i> (inclusive) to <i>j</i> (exclusive)
toLowerCase (), toUpperCase ()	a new String with all lowercase or uppercase letters

## Random Methods (5.1)

nextBoolean ()	random true/false result
nextDouble ()	random real number between 0.0 and 1.0
nextInt ()	random integer
nextInt ( <b>max</b> )	random integer between 0 and <i>max</i>