






Remove an Element from a Max Heap

Now that we can add elements to our heap let's see how we can remove elements. Removing and inserting elements both require similar logic. In a max heap you will usually want to remove the greatest value, so this involves simply extracting it from the root of our tree. This will break the heap property of our tree, so we must reestablish it in some way. Typically, for a max heap this is done in the following way:

1. Move the last element in the heap into the root position.
2. If either child of the root is greater than it, swap the root with the child of greater value.
3. Continue swapping until the parent is greater than both children or you reach the last level in the tree.

Instructions: Add a method to our max heap called `remove`. This method should return the greatest value that has been added to our max heap and remove it from the heap. It should also reorder the heap so the heap property is maintained. After removing an element, the next greatest element remaining in the heap should become the root.

	The <code>MaxHeap</code> data structure should exist.
	<code>MaxHeap</code> should have a method called <code>print</code> .
	<code>MaxHeap</code> should have a method called <code>insert</code> .
	<code>MaxHeap</code> should have a method called <code>remove</code> .
	The <code>remove</code> method should remove the greatest element from the max heap while maintaining the max heap property.