Given the String, "OUT OF ORDER", show the steps/swaps that would occur during the described sort if we turned the string into a char array. Assume we compare by alphabetical order and that spaces (_) are the smallest.

Insertion Sort
OUT_OF_ORDER

Walking the halls of Soda late at night, Prof. Hilfinger came upon a student in distress. A deadline was fast approaching, and after three all-nighters the exhausted student had dropped his only remaining coins into the soda machine without first noticing the paper sign taped to it. The sign read OUT OF ORDER in large red capitals.

"Can you fix it?" asked the student, who knew of Hilfinger's skill with primitive machines.

Hilfinger studied the machine for a moment, then crossed out the letters of the sign and under them wrote DEFOOORRTU. Immediately a soda can dropped into the chute.

Satisfied, Hilfinger continued on his way.

Selection Sort
OUT_OF_ORDER

What type of arrays would insertion sort be fast at?

You are the proud owner of Berkeley Bytes Buffet and business is doing well! You currently have the policy that children under 8 eat for free and senior citizens eat for 50 percent off. Since you're the savvy business owner that you are, you keep track of all the ages of your customers. Summer is coming around and you want to see how much of each age which is an integer is coming to your buffet to plan out your marketing.

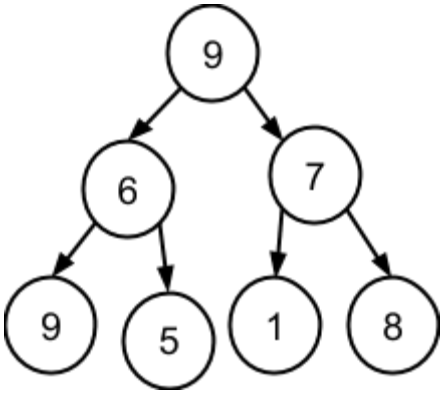
a. **Write an algorithm that takes in a int array of all customers and returns a histogram (how much of each age) as an array of size 150 which we assume is the max age.**
b. **Now for tax reasons, you need to submit a list of the ages of all your customers in sorted order. Write an algorithm that takes an int array of all customers and returns a sorted array. HINT: This can be done in linear time**

```
class CountSort {
      static int maxAge = 150;

      // Part A
      public static int[] histOfAges(int[] ages) {









      }
```

```
	// Part B
	public static int[] ageSort(int[] ages) {









	}
	public static void main(String[] args) {
		int[] test1 = {10,9,8,5,1,4,5,6,7,7,7,40,13,21,64,94,33};
		int[] rtn = ageSort(test1);
		System.out.println(Arrays.toString(rtn));
	}
}
```

Linear time sorting is great, but we can't use it in the general case. In what other cases would we be able to use this type of sort? Can you identify the necessary conditions?

(min)heapify this!



For reference: https://www.cs.usfca.edu/~galles/visualization/Heap.html

Now Insert a 2

Remove the 1

Write HeapSort, which takes in an unsorted array and uses a Heap to solve this problem. How long will this run?

```
public interface Heap<T> {
      Heap(T[] arr); // Heapify!
      void add(T item);
      T removeSmallest();
      boolean isEmpty();
}
```

You can use “Heap<Integer> h = new Heap<Integer>(ints);” to create a new min heap out of an array of ints.

```
public static List<T> heapSort(List<T> unsorted) {









}
```