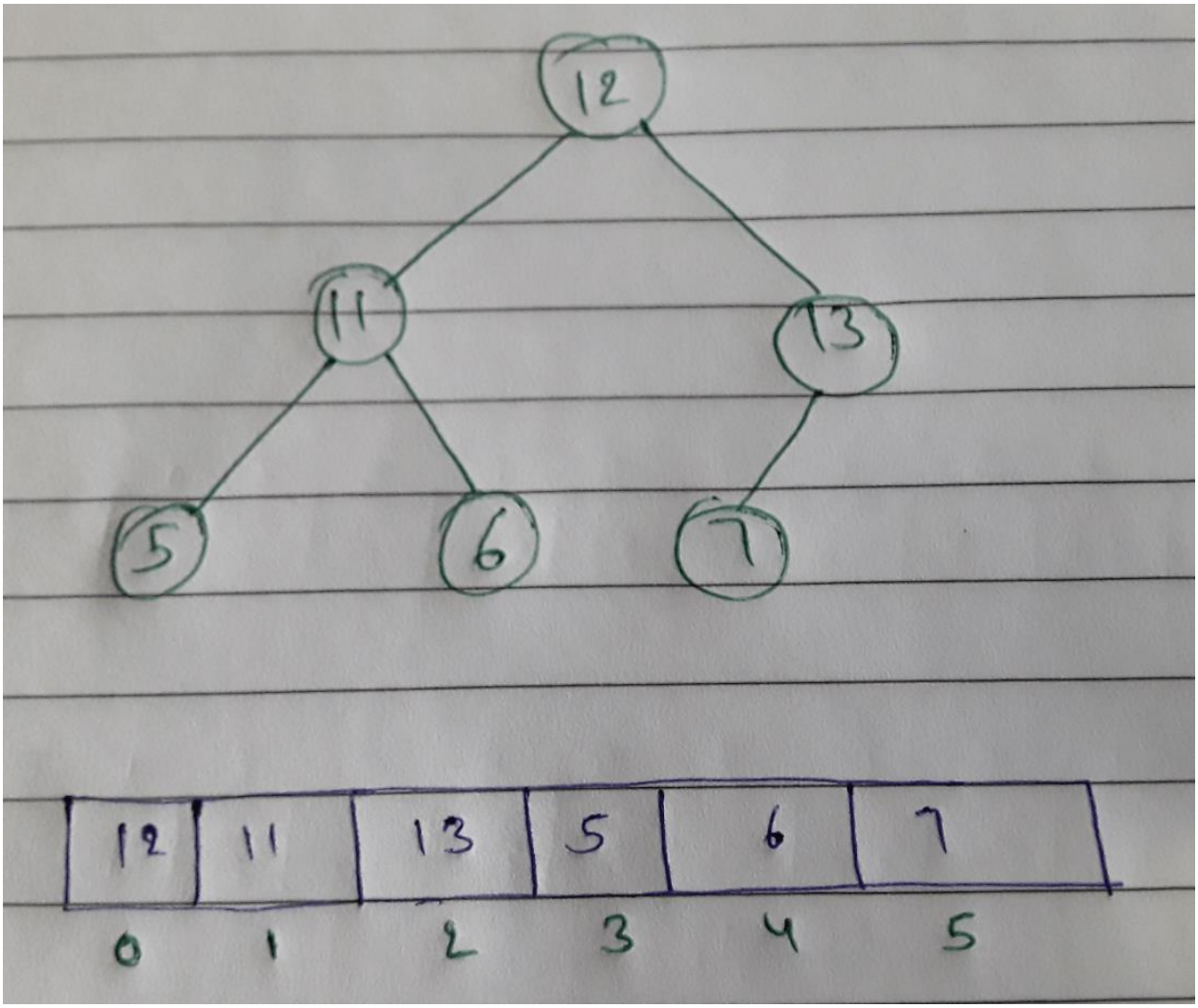
Heap Sort

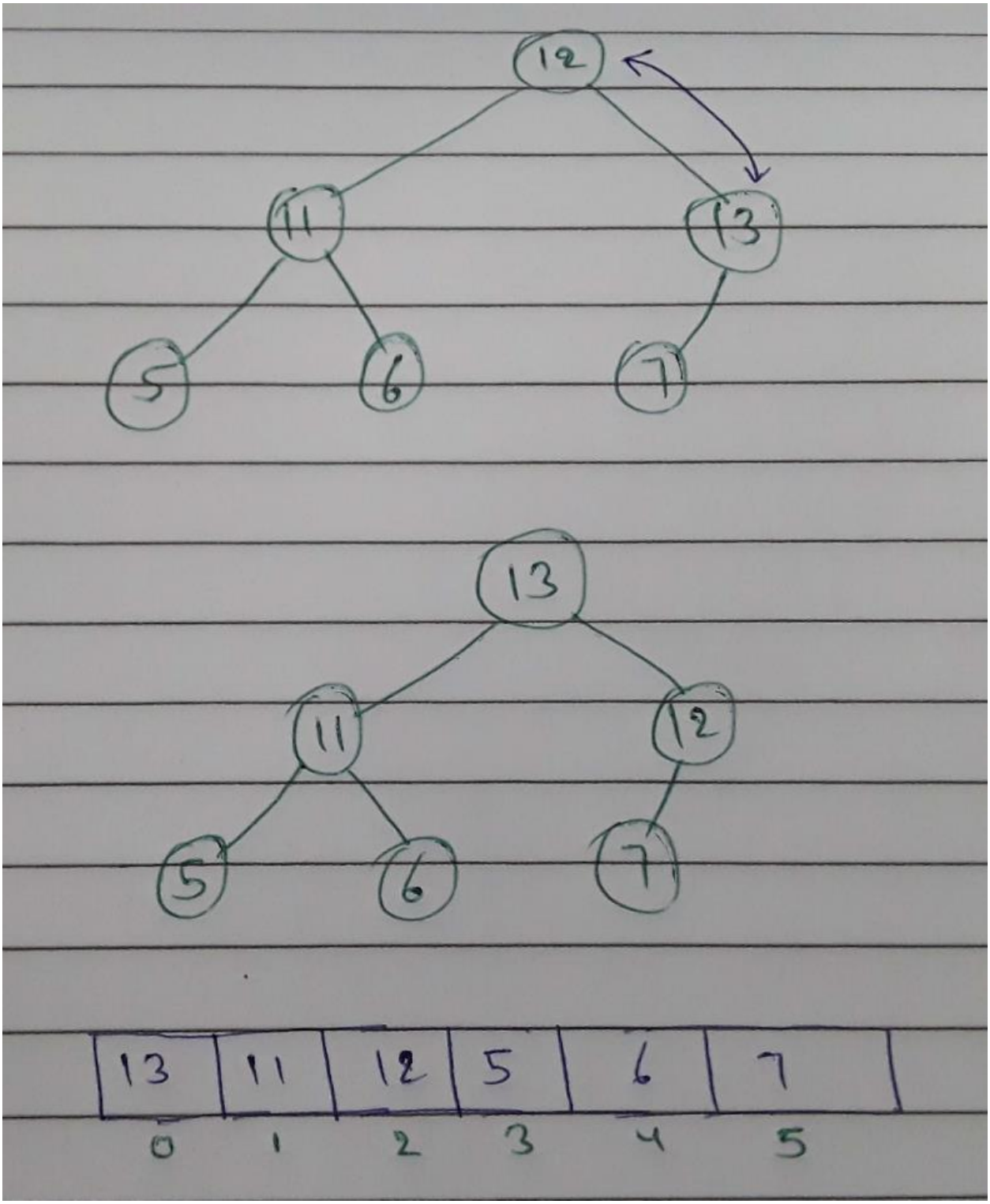
1. Build a Max Heap from given array elements (using Heapify operation)
2. Delete element from Heap one by one (using Heapify operation)

So how does this sort the given input array?

e.g. [12, 11, 13, 5, 6, 7]



Building Max Heap from given array elements



Heapify Operation

```
public static void heapify(int arr[], int n, int i) {
		int largest = i; // Initialize largest as root
		int l = 2 * i + 1; // left = 2*i + 1
		int r = 2 * i + 2; // right = 2*i + 2

		// If left child is larger than root
		if (l < n && arr[l] > arr[largest])
			largest = l;

		// If right child is larger than largest so far
		if (r < n && arr[r] > arr[largest])
			largest = r;

		// If largest is not root
		if (largest != i) {
			int swap = arr[i];
			arr[i] = arr[largest];
			arr[largest] = swap;

			// Recursively heapify the affected sub-tree
			heapify(arr, n, largest);
		}
}
```

Building the Max Heap

```
public static void buildHeap(int arr[]) {
		int n = arr.length;
		// Build heap (rearrange array)
		for (int i = n / 2 - 1; i >= 0; i--)
			heapify(arr, n, i);
}
```

Deleting Elements

```
public static void heapsort(int arr[]) {
		buildHeap(arr);

		int n = arr.length;
		// One by one delete an element from heap
		for (int i = n - 1; i >= 0; i--) {
			// Move current root to end
			int temp = arr[0];
			arr[0] = arr[i];
			arr[i] = temp;
			// call heapify on the reduced heap
			heapify(arr, i, 0);
		}
}
```