6.4-2.

Argue the correctness of HEAPSORT using the following loop invariant:

> At the start of each iteration of the **for** loop of line 2–5, the subarray $A[1\,..\,i]$ is a max-heap containing the i smallest elements of $A[1\,..\,n]$, and the subarray $A[i+1\,..\,n]$ contains the $n-i$ largest elements of $A[1\,..\,n]$, sorted.

Proof.

Initialization. Prior to the first iteration of the loop, $i = n$. The subarray $A[1\,..\,i]$ is exactly $A[1\,..\,n]$, which is the max-heap established by BUILD-MAX-HEAP, and $A[i+1\,..\,n]$ is an empty array, thus sorted.

Maintenance. To see that each iteration maintains the loop invariant, observe that elements of $A[i+1\,..\,n]$ are successively extracted from the root of max-heap, and as the max-heap property suggests, they are sorted and greater than any element in $A[1\,..\,i]$. Inside the max-heap $A[1\,..\,i]$, $A[1]$ is the largest element. The exchanging operation followed by a decrement on the size of the heap helps append the largest element of the heap to the sorted array, making the sorted array expands from $A[i+1\,..\,n]$ to $A[i\,..\,n]$. Calling to MAX-HEAPIFY on the new heap and decrementing i in the **for** loop update reestablishe the loop invariant for the next iteration.

Termination. At termination, $i = 1$. By the loop invariant, $A[1]$ is the only max-heap containing the smallest elements of $A[1\,..\,n]$, and the subarray $A[2\,..\,n]$ contains the $n-1$ largest elements of $A[1\,..\,n]$, sorted. □

Email address: informlarry@gmail.com