
Assignment 3

CS 514 – Algorithms

Submitted By: Aman Pandita

Onid: panditaa@oregonstate.edu

LOOP INVARIANT:

At the start of each iteration of the for loop, 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.

1. Initialization:

The algorithm creates a max heap from the array $A[1.....n]$ before the second “for loop” in the “heapsort” function starts. Since the sorting hasn’t begun yet, it holds that the subarray $A[1....i]$ is a max-heap for any i at this time since the entire array is a max-heap.

This proves that the requirement of startup has been satisfied.

2. Maintenance:

During each iteration, the algorithm takes the root (the maximum element in the max-heap), swaps it with the element at the last position, and then reduces the heap size by one. The maximum element is now correctly placed at the end of the array. After this, the algorithm calls ‘**heapify**’ on the root to ensure the remaining elements also form a max-heap.

Considering the loop invariant, the subarray $A[1....i]$ is a max-heap, and the elements in the subarray $A[i+1....n]$ are sorted. After each iteration, the largest element of the max-heap is moved to its correct position, and ‘**heapify**’ ensures that the invariant is maintained by making sure that the remaining elements are still in a max-heap structure. Hence, the maintenance condition is satisfied.

3. Termination:

The loop terminates when all elements have been put in their correct positions, meaning only one element remains in the max-heap. At this point, since every iteration places the maximum element of the remaining max-heap in its correct position, and the loop invariant has been maintained throughout the algorithm, we are left with a sorted array. Thus, the termination condition is met, and the algorithm is correct.