Advanced Data Structures and Algorithms

Comprehensive Assignment Solutions

Arya Raag
Roll No: A125002

M.Tech (Computer Science and Engineering)

January 5, 2026

1 Heap Operations and Analysis

Question 2. In an array of size n representing a binary heap, prove that all leaf nodes are located at indices from $\lfloor \frac{n}{2} \rfloor + 1$ to n.

Detailed Solution:

Properties of Array-Based Heaps

A binary heap is typically implemented using an array $A[1 \ldots n]$. For any node located at index i:

- The index of the **Parent** is $\lfloor i/2 \rfloor$.
- The index of the **Left Child** is $2i$.
- The index of the **Right Child** is $2i + 1$.

Definition of a Leaf Node

A node is defined as a *leaf* if it has no children. In the context of the array representation, a node at index i is a leaf if and only if its left child index exceeds the bounds of the array. That is, the node i has no children if:

$$\text{Left}(i) > n$$

Substituting the formula for the left child:

$$2i > n$$

Derivation of the Index Range

We solve the inequality $2i > n$ for i:

$$i > \frac{n}{2}$$

Since array indices must be integers, the smallest integer i that satisfies strictly $i > n/2$ is:

$$i_{\min} = \left\lfloor \frac{n}{2} \right\rfloor + 1$$

Conversely, the largest possible index in an array of size n is simply n. Therefore, any node with an index i such that:

$$\left\lfloor \frac{n}{2} \right\rfloor + 1 \leq i \leq n$$

does not have a left child (and consequently, cannot have a right child, as heaps are filled from left to right).

Conclusion

We have rigorously shown that the property of being a leaf node corresponds strictly to the index condition $2i > n$. Thus, the leaves of the heap occupy precisely the second half of the array, specifically the range:

$$\left[\left\lfloor \frac{n}{2} \right\rfloor + 1, \ldots, n\right]$$
