MTech CSE – 1st Semester
Student Name: SIDDHARTH KUMAR
Student ID: A125021

Question 3(b)

Using the result from Part 3(a), prove that the time complexity of the BUILD-HEAP algorithm is $O(n)$.

Answer:

Background

The BUILD-HEAP algorithm constructs a binary heap from an unordered array of n elements by repeatedly applying the HEAPIFY operation to internal nodes, starting from the lowest level and moving upward toward the root.

A key fact used in this analysis is the result from Part (a), which states that the number of nodes at height h in a heap of size n is at most:

$$\left\lceil \frac{n}{2^{h+1}} \right\rceil .$$

Cost of Heapify at Height h

The running time of the HEAPIFY operation on a node is proportional to the height of that node, since Heapify may move down the heap by at most h levels.

Thus, the time required to Heapify a node of height h is:

$$O(h).$$

Total Cost of Build-Heap

To compute the total running time of BUILD-HEAP, we sum the cost of Heapify over all nodes, grouped by their heights.

Let:

- h denote the height of a node,
- the number of nodes at height h be at most $\left\lceil \frac{n}{2^{h+1}} \right\rceil$,
- the cost of Heapify at height h be $O(h)$.

The total cost $T(n)$ can be bounded as:

$$T(n) \leq \sum_{h=0}^{\lfloor \log n \rfloor} \left(\frac{n}{2^{h+1}} \cdot O(h) \right).$$

Simplifying the Summation

Factoring out n, we obtain:

$$T(n) = O\left(n \sum_{h=0}^{\lfloor \log n \rfloor} \frac{h}{2^{h+1}} \right).$$

The series:

$$\sum_{h=0}^{\infty} \frac{h}{2^{h+1}}$$

is a convergent series whose value is a constant.

Therefore, the summation is bounded above by a constant independent of n.

Final Time Complexity

Substituting this result back into the expression for $T(n)$:

$$T(n) = O(n).$$

Conclusion

Although a single call to HEAPIFY may take $O(\log n)$ time, most nodes in the heap are located near the leaves and have very small height. Only a few nodes near the root have large height and incur higher cost.

As a result, the total cost of building a heap from an array of n elements is:

$$\boxed{O(n)}.$$

This analysis explains why the BUILD-HEAP algorithm runs in linear time, despite using the HEAPIFY operation internally.