The Time Complexity of Building a Heap

Abstract

A derivation of the time complexity for building a heap using an arithmetico-geometric series.

BUILD-HEAP calls MAX-HEAPIFY on all nodes except for those in the last layer.

Note that, in a full binary tree of height h, there are $n = 2^{h+1} - 1$ nodes.

You may assume, as we are running MAX-HEAPIFY, an $O(\log n)$ algorithm, roughly n times, that the time complexity of BUILD-HEAP must be $O(n \log n)$.

We can achieve a better upper bound than this, as MAX-HEAPIFY's time complexity depends directly on the *remaining* height of the tree.

This matters, as nodes *one* layer before the leaves have a worst case of 1 swap, for nodes *two* layers before the leaves, it is 2 swaps, and so on, but, our previous naive calculation assumed that all nodes would all have a worst case of h swaps.

The worst case number of swaps, for a full binary tree of height h, can be defined by:

$$T(n) = \sum_{i=0}^{h-1} 2^i (h - i) \quad \text{where } n = 2^{h+1} - 1$$

This can be interpreted as:

> For every height, h, starting from the root layer ($i = 0$), to one layer before the leaves, ($i = h - 1$), each node (2^i nodes on layer i) will at worst traverse the full remainder of the tree (that being $h - i$)

Expanding the sum, we may write:

$$T(n) = h \sum_{i=0}^{h-1} 2^i - \sum_{i=1}^{h-1} i 2^i$$

Note the lower bound of the second summation changes, as at $i = 0$, the value is zero, so it contributes nothing to the sum.

The first sum is a simple geometric series, the second is an arithmetico-geometric series

$$S = \sum_{i=1}^{h-1} i2^i$$

$$\begin{aligned} S &= 1 \cdot 2^1 + 2 \cdot 2^2 + 3 \cdot 2^3 + \cdots + (h-2)2^{h-2} + (h-1)2^{h-1} \\ 2S &= \qquad\quad 1 \cdot 2^2 + 2 \cdot 2^3 + 3 \cdot 2^4 + \cdots + (h-2)2^{h-1} + (h-1)2^h \end{aligned}$$

Each power of 2 in $2S$ (except the last) has a corresponding term in S. As such, we subtract $2S$ from S. The telescoping cancellation leaves:

$$\begin{aligned} S - 2S &= \underbrace{2 + 2^2 + 2^3 + \cdots + 2^{h-2} + 2^{h-1}}_{\text{geometric series}} - (h-1)2^h \\ -S &= (2^h - 2) - (h-1)2^h \end{aligned}$$

Further simplifying, we obtain our closed form for the summation.

$$\boxed{\sum_{i=1}^{h-1} i2^i = 2^h(h-2) + 2}$$

Substituting the closed forms for the series, and simplifying

$$\begin{aligned} h(2^h - 1) - 2^h(h-2) - 2 \\ = 2^{h+1} - h - 2 \end{aligned}$$

As $n = 2^{h+1} - 1$, we may substitute n into the formula.

$$T(n) = n - \log_2(n+1)$$

This is clearly $O(n)$.