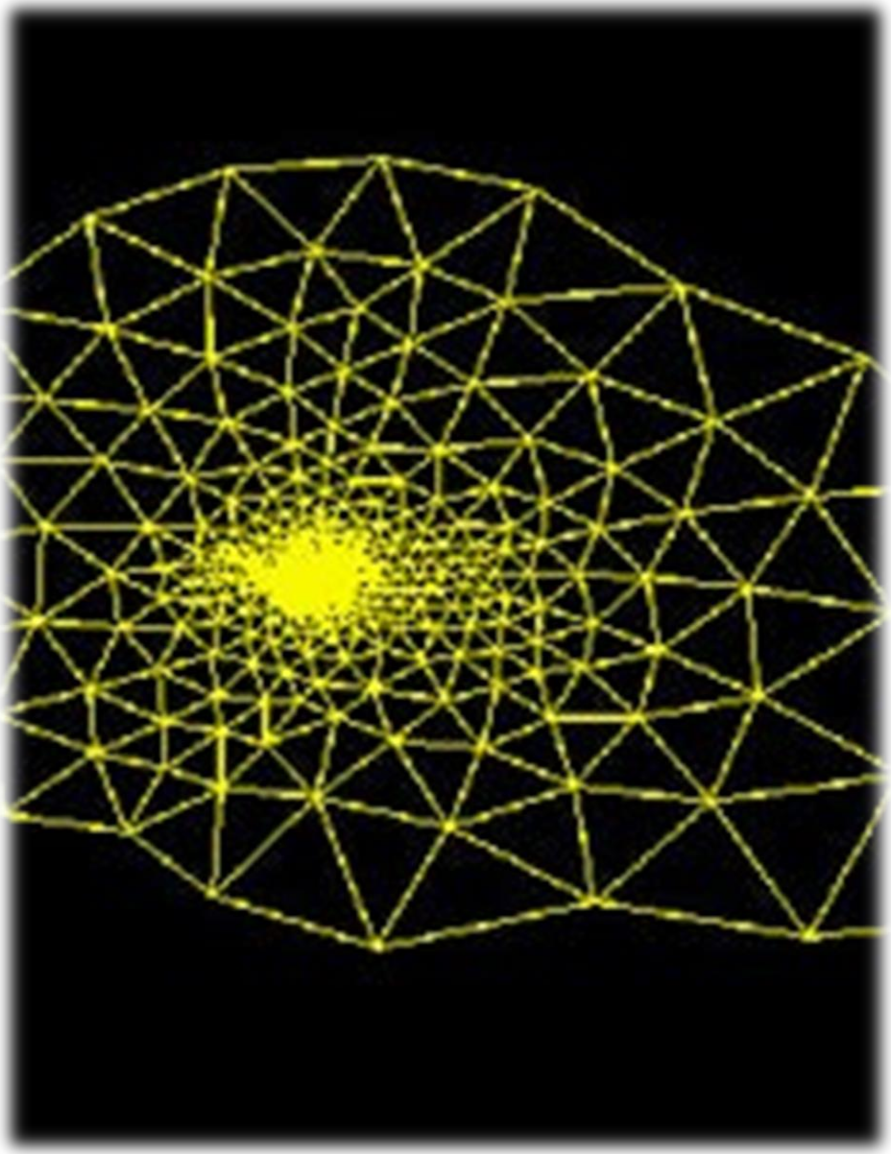
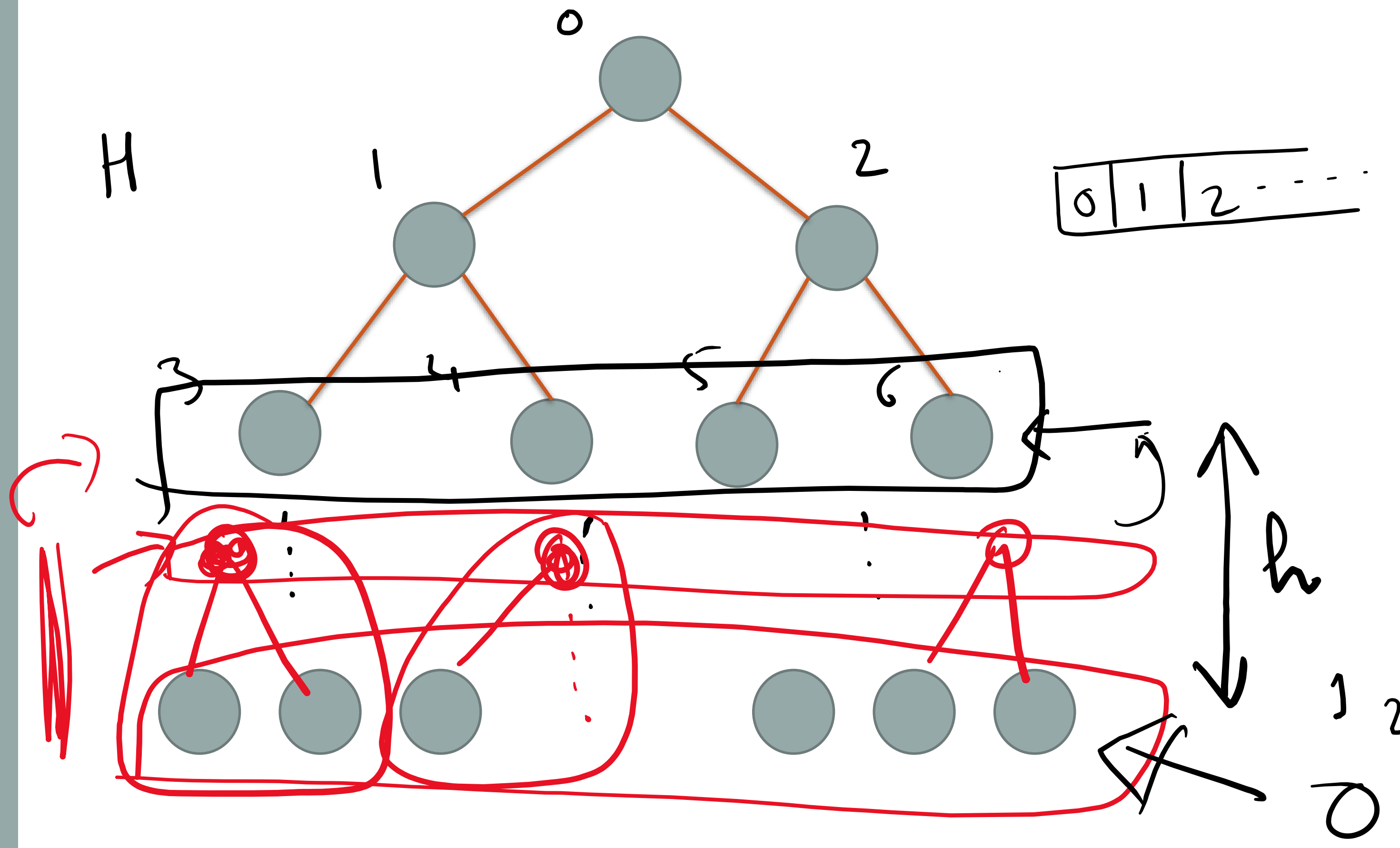
Data Structures & Algorithms

Build Heap Time Complexity



Build Heap from array A

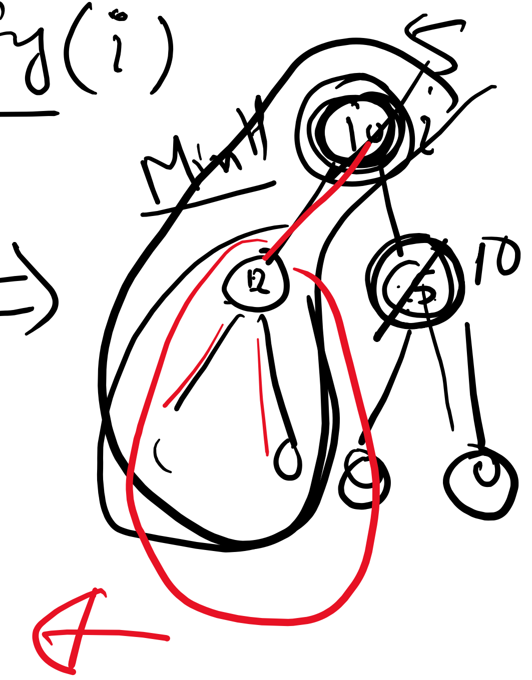
BuildHeap()



H. Heapify(i)

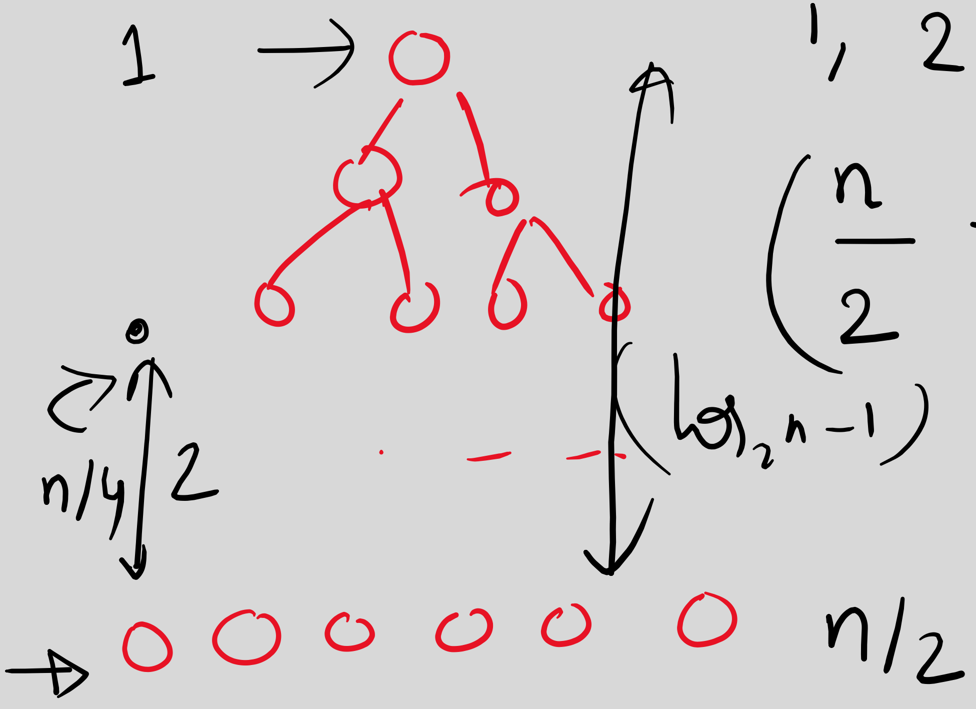
log(n)

⇒



$\log n \times \frac{n}{2}$

$O(n\log n)$



1, 2, 4, 8, ...

$$\left(\frac{n}{2} + 1\right)$$

$$\cancel{\frac{n}{2}\times 0} \quad \left(\frac{n}{2^2}\times 1\right) + \left(\frac{n}{2^3}\times 2\right) + \left(\frac{n}{2^4}\times 3\right)$$

$$+ \cdots\cdots + \frac{n}{2^{\log_2 n}} \times \left(\log_2 n - 1\right)$$

$$2^{\log_2 n} = n$$

$$T = \frac{n}{2^2}\times 1 + \frac{n}{2^3}\times 2 + \cdots \qquad \frac{n}{2^{\log n}}\left(\log n - 1\right)$$

T

$$T = \frac{n}{2^2}\times 1 + \frac{n}{2^3}\times 2 + \frac{n}{2^4}\times 3 + \cdots + \frac{n}{2^{\log n}}\times(\log n - 1)$$

$$\boxed{S = \frac{a(1-r^n)}{1-r}}$$

$$\frac{T}{2} = \frac{n}{2^3}\times 1 + \frac{n}{2^4}\times 2 + \cdots + \frac{n}{2^{\log n}}\times(\log n - 2) + \frac{n}{2^{\log n+1}}(\log$$

$$(\log n - 1)$$

$$\frac{T}{\cancel{2}} = \left[\frac{n}{2^2} + \frac{n}{2^3} + \frac{n}{2^4} + \cdots \frac{n}{2^{\log n}}\right] + \cancel{\frac{n}{2^{\log n + 1}}}(\log n - 1)$$

$$T = \frac{n}{2}\left[1 + \frac{1}{2} + \frac{1}{2^2} + \cdots \frac{1}{2^{\log n - 2}}\right] + {}''$$

$$= \frac{n}{\cancel{2}}\left(\frac{1 - \frac{2}{2^{\log n}}}{\cancel{1}}\right) + n \, \cancel{\log n} \, C$$

$$= n\left(\frac{n-2}{n}\right)$$

$$T = \cancel{n}\left(\frac{n-2}{\cancel{n}}\right) + C$$

$$= n + C$$

$$\approx \boxed{\underline{\underline{O(n)}}}$$

$O(n \log n)$

$\underline{\underline{\Sigma \text{ GP}}}$