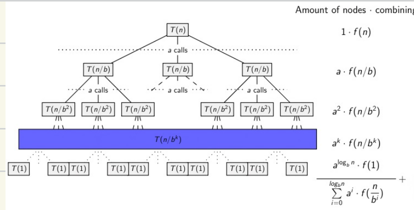
1.1 Master Method

Recurrence equation:

for $a \geq 1$, $b \geq 2$: $T(n) = \begin{cases} \Theta(1) & \text{if } n = 1 \\ aT(n/b) + f(n) & \text{else} \end{cases}$



time in leaves: $\Theta(a^{\log_b n}) = \Theta(n^{\log_b a})$

tree: $T(n) = \Theta(n^{\log_b a}) + \sum_{i=0}^{\log_b n} a^i f\left(\frac{n}{b^i}\right)$

work in leaves ↙ ↳ work in combining results

1) Leaves: $f(n) \in O(n^{\log_b a - \varepsilon}) \Rightarrow T(n) \in \Theta(n^{\log_b a})$, $\varepsilon > 0$

2) Even: $f(n) \in \Theta(n^{\log_b a}) \Rightarrow T(n) \in \Theta(n^{\log_b a} \log n)$

3) Root: $f(n) \in \Omega(n^{\log_b a + \varepsilon}) \Rightarrow T(n) \in \Theta(f(n))$, $\varepsilon > 0$

iff $f(n)$ - polynomial or $\exists c < 1$ s.t. $af\left(\frac{n}{b}\right) \leq cf(n)$

1.2. Practice

Merge sort: $T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$

$\Rightarrow a = 2$, $b = 2 \Rightarrow n^{\log_b a} = n$ (work in leaves)

$f(n) \in \Theta(n)$ (work in root)

$\Rightarrow$ Even: $T(n) \in \Theta(n \log n)$

$T(n) = 9T\left(\frac{n}{3}\right) + n$

$a = 9$, $b = 3$, $n^{\log_b a} = n^2$ (leaves)

$f(n) = n$ (root)

$\Rightarrow$ Leaves: $T(n) \in \Theta(n^2)$

```
function BINARY-SEARCH(A,p,q,r,s)
    q ← (p + r)/2
    if p > r then
        return false
    else if a[q] = s then
        return q
    else if a[q] > s then
        return BINARY-SEARCH(a,p,q-1,s)
    else
        return BINARY-SEARCH(a,q+1,r,s)
```

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

$a = 1$, $b = 2$, $n^{\log_b a} = 1$ (leaves)

$f(n) \in \Theta(1)$ (root)

$\Rightarrow$ Even: $T(n) \in \Theta(\log n)$

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

$n^{\log_b a} = 1$ (leaves)

$f(n) = n$ (root)

$\Rightarrow$ Root: $T(n) \in \Theta(n)$

($f(n)$ - polynomial)

2. Closest pair of points

set of points S, $\forall$ point $p_i = (x_i, y_i)$

distance $d(i,j) = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2}$

Find pair i, j s.t. $d_{min}(i,j)$.

Divide: vertical line L, split points in half

Conquer: closest pair on each side

Combine: closest pair with one point on each side

Return minimum of the three

δ - minimum of left-right smallest distance

combine: consider points in $[L-\delta, L+\delta]$

if all points in $\Rightarrow$ M- horizontal line

s_i - point in 2δ-strip with i^{th} smallest y-coor

$|i-j| > 11 \Rightarrow d(s_i, s_j) \geq \delta$

Sort points by x-coordinate

function Closest-Pair(p1, … , p2)

    if n=1 then

        return INF

    L <- median x-coordinate

    d1 <- Closest-Pair(Points left of L)

    d2 <- Closest-Pair(Points right of L)

    d <- min(d1,d2)

    get list of points within d from L.

    sort list by y-coordinate.

    Scan by y-order, compare point to next 11, update d

    return d

$$T(n) = 2T\left(\frac{n}{2}\right) + O(n\log n) \in \Theta(n\log n)$$