Ozan Argit Önçeken
1901042259

1) $T(n) = aT\left(\frac{n}{b}\right) + n^k$ $\quad f(n) = n^k$

$$\left.\begin{array}{ll} \Theta(n^{\log_b a}) & f(n) = O(n^{\log_b a - \varepsilon}) \\ \Theta(n^{\log_b a}\log n) & f(n) = \Theta(n^{\log_b a}) \\ \Theta(f(n)) & f(n) = \Omega(n^{\log_b a + \varepsilon}) \wedge \\ & af(n/b) < cf(n) \end{array}\right\} \begin{array}{l}\varepsilon > 0 \\ c < 1\end{array}$$

a) $a = 16 \quad b = 4 \quad f(n) = n!$

$$\Theta(n^{\log_4 16}) = \Theta(n^2)$$

$$\lim_{n\to\infty}\left(\frac{n!}{n^2}\right) = \lim_{n\to\infty}\left(\frac{n(n-1)(n-2)!}{n^2}\right) = \lim_{n\to\infty}\frac{(n^2-n)(n-2)!}{n^2} =$$

$$\lim_{n\to\infty}\frac{(n^2-n)}{n^2}\cdot\lim_{n\to\infty}(n-2)! = \lim_{n\to\infty}\frac{n^2}{n^2}\cdot\lim_{n\to\infty}(n-2)! = \lim_{n\to\infty}(n-2)! = \infty$$

$\Rightarrow f(n) \in \Omega(n^2)$

$16f(n/4),$

$16\cdot\left(\frac{n}{4}\right)! < cf(n)$ ✓ $\qquad \underline{\Theta(n!)}$

b) $a = \sqrt{2} \quad b = 4 \quad f(n) = \log n$

$$\Theta(n^{\log_4 2^{1/2}}) = \Theta(n^{1/4})$$

$$\lim_{n\to\infty}\frac{n^{1/4}}{\log(n)} = \lim_{n\to\infty}\frac{n^{1/4}}{\ln(n)} \overset{\text{L'Hopital}}{=} \lim_{n\to\infty}\left(\frac{\frac{1}{4n^{3/4}}}{\frac{1}{n}}\right) = \lim_{n\to\infty}\left(\frac{1}{4n^{3/4}}\cdot n\right)$$

$$= \lim_{n\to\infty}\left(\frac{n^{1/4}}{4}\right) = \infty \Rightarrow f(n) \in \Omega(\log(n))$$

$af(n/b),$

$\sqrt{2}\cdot\log\left(\frac{n}{4}\right) < c\cdot\log n$ ✓ $\qquad \underline{\Theta(\log(n))}$

c) $a=8 \quad b=2 \quad f(n)=4n^3$

$\Theta(n^{\log_2 8}) = \Theta(n^3) \quad \lim_{n\to\infty} \frac{n^3}{4n^3} = \lim_{n\to\infty} \frac{1}{4} = \frac{1}{4} \Rightarrow f(n) = \Theta(n^{\log_2 8})$

$\Rightarrow \Theta(n^{\log_2 8} \cdot \log(n)) = \Theta(n^3 \cdot \log(n))$

d) $a=64 \quad b=8 \quad f(n) = -n^2 \cdot \log(n)$

Master theorem cannot apply, f(n) is negative

e) $a=3 \quad b=3 \quad f(n)=\sqrt{n}$

$\Theta(n^{\log_3 3}) = \Theta(n) \quad \lim_{n\to\infty} \frac{n}{\sqrt{n}} = \lim_{n\to\infty} n^{1/2} = \infty \Rightarrow$

$f(n) = O(\sqrt{n})$

$\Theta(n^{\log_3 3}) = \Theta(n)$

f) $a=2^n \quad b=2 \quad f(n) = -n^n$

Master theorem does not apply because

* a is not constant
* f(n) is negative

g) $a=3 \quad b=3 \quad f(n) = \frac{n}{\log(n)}$

Master theorem does not apply because non-polynomial difference between f(n) and $n^{\log_b a}$.

2)

$\Theta(n^{\log_b a}) \quad f(n) = O(n^{\log_b a - \varepsilon})$

$\Theta(n^{\log_b a} \cdot \log(n)) \quad f(n) = \Theta(n^{\log_b a - \varepsilon})$

$\Theta(f(n)) \quad f(n) = \Omega(n^{\log_b a + \varepsilon})$

$af(n/b) \leq c f(n)$

$\varepsilon > 0$

$c < 1$

a) $T(n) = 9T(n/3) + n^2$

$a=9 \quad b=3 \quad f(n)=n^2$

$\Theta(n^{\log_3 9}) = \Theta(n^{2\log_3 3}) = \Theta(n^2) = f(n) \Rightarrow$

$\Theta(n^2) = O(n^2)$

b) $T(n) = 8T(n/2) + n^3$

$a=8 \quad b=2 \quad f(n)=n^3$

$\Theta(n^{\log_2 8}) = \Theta(n^{3\log_2 2}) = \Theta(n^3) = f(n) \Rightarrow$

$\Theta(n^3) = O(n^3)$

c) $T(n) = 2T(n/4) + \sqrt{n}$

$a=2 \quad b=4 \quad f(n)=\sqrt{n}$

$\Theta(n^{\log_4 2}) = \Theta(n^{1/2 \cdot \log_2 2}) = \Theta(\sqrt{n}) = f(n) \Rightarrow$

$\Theta(\sqrt{n}) = O(\sqrt{n})$

For choose:

$O(\sqrt{n})$ because

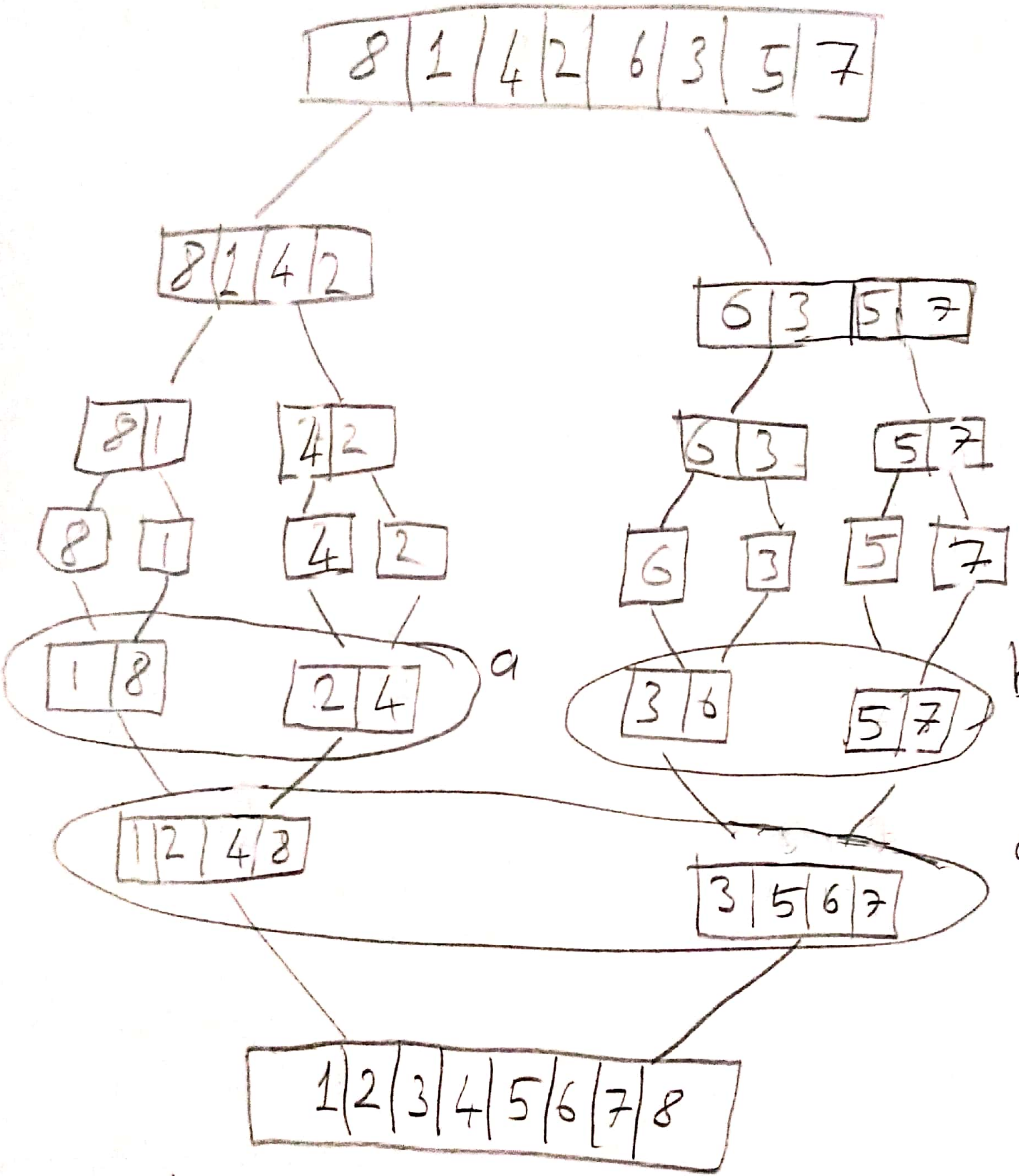
$\lim_{n\to\infty} \frac{\sqrt{n}}{n^2} = \lim_{n\to\infty} \frac{1}{n^{3/2}} = 0 \Rightarrow O(n^2) > O(\sqrt{n})$

$\lim_{n\to\infty} \frac{n^3}{n^2} = \lim_{n\to\infty} n = \infty \Rightarrow O(n^3) > O(n^2) \Rightarrow$

$O(n^3) > O(n^2) > O(\sqrt{n})$

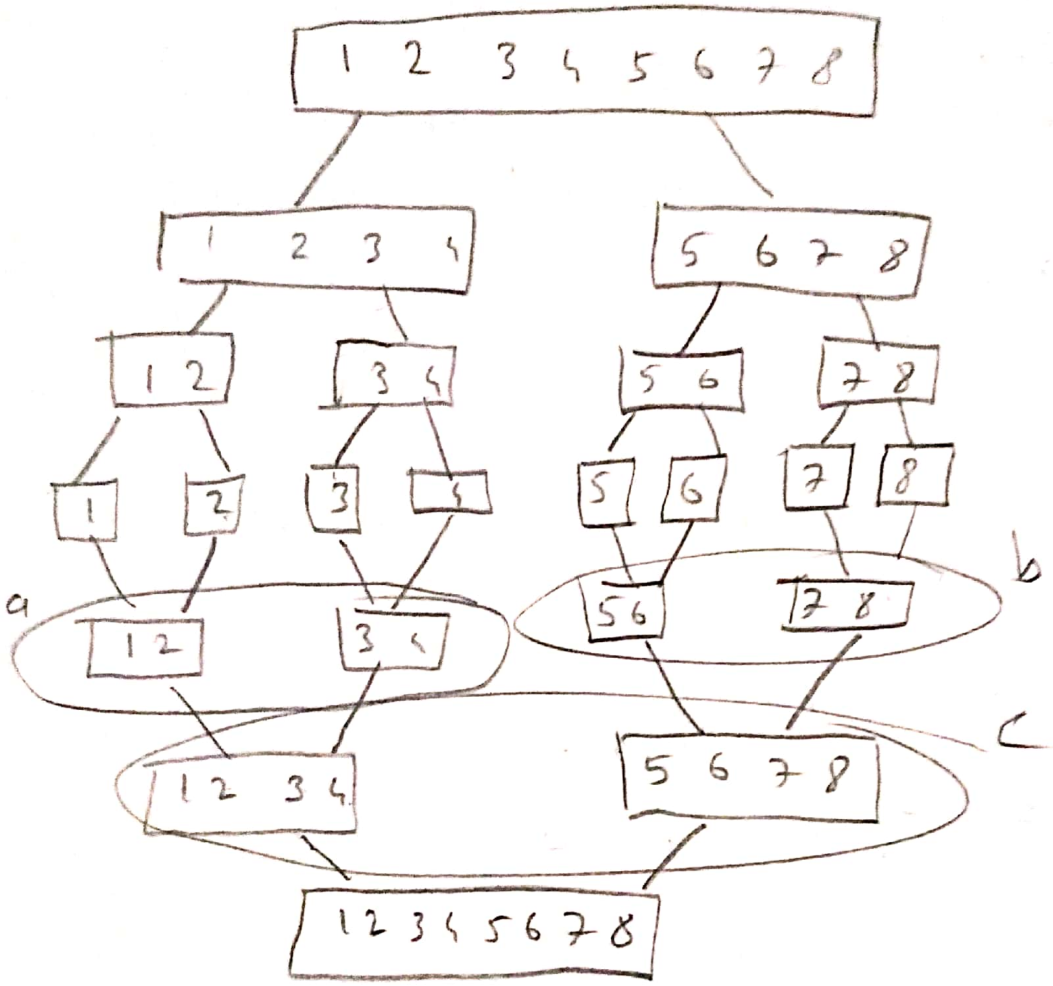
3) a) i) [8, 1, 4, 2, 6, 3, 5, 7] because

CamScanner ile tarandı



For doing comparison a, 1 compares itself with 3 and gets first place 8 compares itself with 2 and 4. (3 comparison occured that place is maximum), b is same with a. At c comparison 1 compares with 3, puts itself at first place, 2 compares itself with 3. 4 compares itself with 3 and 5, 8 compares itself with 5, 6 and 7. The right side automatically puts themselves at the right place between 4 and 8. (6 comparison is making)

ii) [1, 2, 3, 4, 5, 6, 7, 8] because



At comparison a 1 compares itself with 3 and gets first place. 2 compares itself and gets second place (2 comparison is minimum) b is same as a. At comparison C 1, 2, 3 and 4 compares themselves with 5 and rest of automatically gets the right place (4 comparison occured)

b) i) [4, 6, 8, 7, 5, 2, 3, 1] because if middle point is at starting the rest of should swap locations. After that if the second half comes, they should swap locations too. After 4 I choose 6 because 6 is the middle of second half, after that 8 and 7 because they will go right of 6 and changing their places too (5 will goes left of 6). After 5 I chose 2 because 2 is middle of first half. 3 and 1 are consecutive because if it was 1 and 3 after setting 2 at the right place 1 and 3 sets right place automatically.

ii) [1, 2, 3, 4, 5, 6, 7, 8] because no swap needed array sorted correctly.

4) $T(n) = 2T(n/2) + 1 \Rightarrow T(n/2) = 2 \cdot T(n/4) + 1$

$$T(n) = 2(2 \cdot T(n/4) + 1) + 1 = 4 \cdot T(n/4) + 2 + 1$$

$$= 4(2 \cdot T(n/8) + 1) + 2 + 1 = 8T(n/8) + 4 + 2 + 1$$

$$= 8(2T(n/16) + 1) + 4 + 2 + 1 = 16T(n/16 + 8 + 4 + 2 + 1$$

$$\vdots$$

$$T(n) = 2^k \cdot T(n/2^k) + 2^{k-1} + \dots + 1$$

$$= 2^k \cdot T(n/2^k) + \frac{2^{k-1} \cdot (2^{k-1} + 1)}{2}$$

$$= 2^k \cdot T(n/2^k) + \frac{2^{2k-2} + 2^{k-1}}{2}$$

$$T(n) = 2^k \cdot T(n/2^k) + 2^{2k-3}$$

5) a)
```
int first ( arr, low, high, x, n) {
    if(high>=low){
        int mid= low+(low+high)/2
        if((mid==0 or x>arr[mid-1]) and (arr[mid]==x))
            return mid;
        if (x> arr[mid])
            return first(arr, mid+1, high, x, n)
        return first(arr, low, (mid-1, x, n)
    }
    //cannot find element
    return -1;
}

void sortI+(arr1, arr2, m, n) {

    int t[m], v[n]
    for (int i=0; i<m; i++){
        t[i] = arr1[i];
        v[i]=0; }
    sort(t, t+m)
    int ind = 0;
    for( int i=0; i<n; i++){ int f = first(t, 0, n-1, A[i], m)
     if(f == -1)
        continue;
     for (int j=f; j<m && t[j] == A[i]); j++){
      visited[j]=1;
     }
} for (int i=0; i<n; i++)
        if(visited[i]==0)
            A[ind++] = t[i]
}
```

CamScanner ile tarandı