Süleyman KORAMAZ
1901042615

CSE 321 Homework-1

① To determine whether $f \in O(g(n))$, $f \in \Omega(g(n))$ or $f \in \Theta(g(n))$:

If $\lim_{n\to\infty} \frac{f(n)}{g(n)} = 0$, $f \in O(g(n))$

$= \infty$, $f \in \Omega(g(n))$

$=$ constant , $f \in \Theta(g(n))$

a) $\lim_{n\to\infty} \frac{2^n}{2^{2n}} = \lim_{n\to\infty} \frac{1}{2^n} = 0 \rightarrow \boxed{f(n) \in O(g(n))}$

b) $\lim_{n\to\infty} \frac{n^2}{n^3} = \lim_{n\to\infty} \frac{1}{n} = 0 \rightarrow \boxed{f(n) \in O(g(n))}$

c) $\lim_{n\to\infty} \frac{3n+1}{2n-5} \overset{\text{L'hospital}}{\Longrightarrow} \lim_{n\to\infty} \frac{3}{2} = \frac{3}{2} \rightarrow \boxed{f(n) \in \Theta(g(n))}$

d) $\lim_{n\to\infty} \frac{4n^2}{n^2} = \lim_{n\to\infty} 4 = 4 \rightarrow \boxed{f(n) \in \Theta(g(n))}$

e) $\lim_{n\to\infty} \frac{\log_2 n}{\log_{10} n} \overset{\text{L'hospital}}{\Longrightarrow} \frac{\frac{1}{n.\ln 2}}{\frac{1}{n.\ln 10}} = \lim_{n\to\infty} \frac{\ln 10}{\ln 2} =$ constant $\rightarrow \boxed{f(n) \in \Theta(g(n))}$

f) $\lim_{n\to\infty} \frac{2^n}{3^n} = \left(\frac{2}{3}\right)^n \rightarrow 0 < \frac{2}{3} < 1 = 0 \rightarrow \boxed{f(n) \in O(g(n))}$

g) $\lim_{n\to\infty} \frac{n^3}{1000n^2} = \lim_{n\to\infty} \frac{n}{1000} = \infty \rightarrow \boxed{f(n) \in \Omega(g(n))}$

h) $\lim_{n\to\infty} \frac{5n+4}{2n+2} \overset{\text{L'hospital}}{\Longrightarrow} \frac{5}{2} \rightarrow \boxed{f(n) \in \Theta(g(n))}$

i) $\lim_{n\to\infty} \frac{\sqrt{n}}{\log_2 n} \overset{\text{L'hospital}}{\Longrightarrow} \lim_{n\to\infty} \frac{\frac{1}{2}\cdot\frac{1}{\sqrt{n}}}{\frac{1}{n.\ln 2}} = \lim_{n\to\infty} \frac{n.\ln 2}{2\sqrt{n}} = \lim_{n\to\infty} \frac{\sqrt{n}.\ln 2}{2} = \infty \rightarrow \boxed{f(n) \in \Omega(g(n))}$

j) $\lim_{n\to\infty} \frac{2^n}{2^{n+1}} = \frac{1}{2} \rightarrow \boxed{f(n) \in \Theta(g(n))}$

(2)

$\lim_{n\to\infty} \frac{\frac{1}{2n}}{\log n} = \lim_{n\to\infty} \frac{1}{2n\cdot\log n} = 0$, so $\boxed{\frac{1}{2n} < \log n}$

$\lim_{n\to\infty} \frac{\log n}{\sqrt{n+5}} \xrightarrow{\text{L'hospital}} \lim_{n\to\infty} \frac{\frac{1}{n\cdot\ln 10}}{\frac{1}{2}\cdot\frac{1}{\sqrt{n+5}}} = \lim_{n\to\infty} \frac{2\sqrt{n+5}}{n\cdot\ln 10}$

$\xrightarrow{\text{L'hospital}} \lim_{n\to\infty} \frac{\frac{1}{\sqrt{n+5}}}{\ln 2} = 0$, so $\boxed{\log n < \sqrt{n+5}}$

$\lim_{n\to\infty} \frac{\sqrt{n+5}}{n+1} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{\frac{1}{2\sqrt{n+5}}}{1} = 0$, so $\boxed{\sqrt{n+5} < n+1}$

$\lim_{n\to\infty} \frac{n+1}{10^n} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{1}{10^n\cdot\ln 10} = 0$, so $\boxed{n+1 < 10^n}$

$\lim_{n\to\infty} \frac{10^n}{n^2\cdot\log n} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{10^n\cdot\ln 10}{2n\log n + \frac{n^2}{\ln 2 n}}$

$\xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{10^n\cdot\ln^2 10}{2\log n + \frac{3}{\ln 2}} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{10^n\cdot\ln^3 10}{\frac{2}{\ln 2\cdot n}}$

$= \lim_{n\to\infty} \frac{10^n\cdot\ln^3 10\cdot\ln 2\cdot n}{2} = \infty$, so $\boxed{n^2\cdot\log n < 10^n}$

$\lim_{n\to\infty} \frac{n+1}{n^2\log n} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{1}{2n\log n + \frac{n^2}{\ln 2 n}} = 0$, so $\boxed{n+1 < n^2\log n}$

$\lim_{n\to\infty} \frac{n^2\log n}{2^n} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{2n\log n + \frac{2}{\ln 2}}{2^n\cdot\ln 2}$

$\xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{2\log n + \frac{2}{\ln 2} + \frac{1}{\ln 2}}{2^n\cdot\ln^2 2} \xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{\frac{2}{\ln 2\cdot n}}{2^n\cdot\ln^3 2}$

$= \lim_{n\to\infty} \frac{2}{\ln 2\cdot n\cdot 2^n\cdot\ln^3 2} = 0$, so $\boxed{n^2\log n < 2^n}$

$\lim_{n\to\infty} \frac{2^n}{10^n} = \lim_{n\to\infty} \left(\frac{1}{5}\right)^n = 0$, so $\boxed{2^n < 10^n}$

$\lim_{n\to\infty} \frac{10^n}{n!} \xrightarrow{\text{L'hosp.}} \left[n! \cong \sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n \right]$

$\longrightarrow \lim_{n\to\infty} \frac{10^n}{\sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n} = \lim_{n\to\infty} \frac{\left(\frac{10\cdot e}{n}\right)^n}{\sqrt{2\pi n}}$

$\xrightarrow{\text{L'hosp.}} \lim_{n\to\infty} \frac{(10\cdot e)^n}{n^n\cdot\sqrt{2\pi n}} = 0$, so $\boxed{10^n < n!}$

$\lim_{n\to\infty} \frac{n!}{n^{2^n}} = \lim_{n\to\infty} \frac{\sqrt{2\pi n}\cdot\left(\frac{n}{e}\right)^n}{n^{2^n}}$

$= \lim_{n\to\infty} \sqrt{2\pi n}\cdot\left(\frac{n}{e\cdot n^2}\right)^n = \lim_{n\to\infty} \sqrt{2\pi n}\cdot\left(\frac{1}{e\cdot n^2}\right)^n$

$\xrightarrow{\text{L'hospital}} \lim_{n\to\infty} \frac{\frac{2\pi}{2\sqrt{2\pi n}}}{(\ln(en)+1)\cdot(en)^n} = \lim_{n\to\infty} \underbrace{\frac{2\pi}{2\sqrt{2\pi n}\cdot(en)^2\cdot(\ln(en)-1)}}$

$\longrightarrow = 0$, so $\boxed{n! < n^{2^n}}$

Answer:

$$\boxed{\frac{1}{2n} < \log n < \sqrt{n+5} < n+1 < n^2\cdot\log n < 2^n < 10^n < n! < n^{2^n}}$$

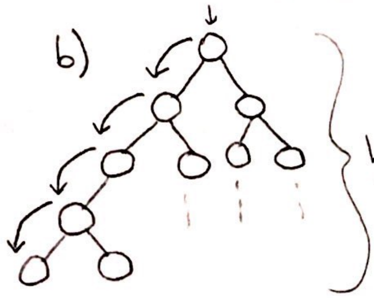
(3) a) There is three process while merging two bst.

NOTE: Height of Binary search trees is "n"
So in worst case # of nodes is "2^n-1"

1- Inorder traversal for create array by using. $\rightarrow (2^n-1)*2$
2- Merge arrays $\rightarrow (2^n-1)*2$
3- Create bst by using array. $\rightarrow (2^n-1)*2$

$6.2^n - 6 \Rightarrow O(2^n)$

b)



height is n.

When we are search smaller element we will travel left side every iteration because binary search tree is sorted. Smaller elements are in the left side always.

In worst case, we travel all layers which equals to height.

$\rightarrow$ Time complexity is $O(n)$

c) There is two process while balancing bst.

NOTE: # of nodes is "2^n-1", in worst case

1- Inorder traversal to store bst in a list. $\rightarrow$ "2^n-1"
2- Buildin tree that creates balanced, binary search tree from list. $\rightarrow$ "2^n-1"

$2.2^n - 2 \Rightarrow O(2^n)$

d) In worst case, value range [x-y] define a range that includes a large portion of the tree and # of nodes is "2^n-1". Time complexity will be $O(2^n)$
We must traversal all the nodes.

NOTE: In my program, range is not importanta for time complexity.
It is always traversal all the nodes.

④ 1. iteration → $i=2$, 2^2+1

2. iteration → $i=2^2+1$, $2^2 = 2^{2^1}$

3. iteration → $i=2^2$, 2^4+1

4. iteration → $i=2^4+1$, $2^4 = 2^{2^2}$

5. iteration → $i=2^4$, 2^8+1

6. iteration → $i=2^8+1$, $2^8 = 2^{2^3}$

7. iteration → $i=2^8$, $2^{16}+1$

8. iteration → $i=2^{16}+1$, $2^{16} = 2^{2^4}$

⋮

kth iteration → result $= 2^{2^{\frac{k}{2}}}$ or $2^{2^{\left(\frac{k+1}{2}\right)}}+1$

$$2^{2^{k/2}} \leq n \longrightarrow \log 2^{2^{k/2}} \leq \log n$$

$$= 2^{k/2} \cdot \log 2 \leq \log n \rightarrow \log 2^{k/2} + \log(\log 2) \leq \log(\log n)$$

$$\Rightarrow \frac{k}{2} \cdot \log_2 2 + \log_2(\log_2 2) \leq \log_2(\log_2 n)$$

$$\Rightarrow \frac{k}{2} + 0 \leq \log_2(\log_2 n)$$

$$\Rightarrow k \leq 2 \cdot \log_2(\log_2 n)$$

→ time complexity is $\boxed{O(\log(\log n))}$

⑤ 1. iteration → %20

2. iteration → %20

3. iteration → %20

⋮

n. iteration → %20

$$\sum_{i=1}^{n} \%20 = n \cdot \frac{2}{10} \Rightarrow \boxed{O(n)}$$

```
function firstEven (L[1:n])
    for i=1 to n do
        if L[i]%2 == 0
            return L[i]
        end if
    end for
    return Ø
end function
```