

1. $T(n) = 2 + 4 + 6 + 8 + 10 + \dots + 2^n$
 Deret = $a \frac{(r^n - 1)}{r - 1} = \frac{2(2^n - 1)}{2 - 1} = 2^{n+1} - 2$

big O = $O(2^n)$

$T(n) \leq C \cdot 2^n$

$2^{n+1} - 2 \leq C \cdot 2^n$

$2 - \frac{2}{2^n} \leq C$

$n=1 \Rightarrow 2 - \frac{2}{2} \leq C$

$C \geq 1$

2. Buktikan bahwa p, q, r positif

$T(n) = pn^2 + qn + r$ adalah $O(n^2) \cap (n^2) \theta(n^2)$

→ Pembuktian Big-O

$T(n) \leq C \cdot f(n)$

$pn^2 + qn + r \leq C \cdot n^2$

$\frac{pn^2}{n^2} + \frac{qn}{n^2} + \frac{r}{n^2} \leq C$

$p + \frac{q}{n} + \frac{r}{n^2}, n=1 \Rightarrow p, q, r=1$

$1 + \frac{1}{1} + \frac{1}{1} \leq C$

$C \geq 3$

→ Big θ

Karena $\theta(n^2)$ dan $\cap(n^2)$ terbukti

dan berderajat sama maka $\theta(n^2)$ benar

3. Tentukan kompleksitas waktu

→ Big-O = $O(n^3)$

$n^3 \leq C \cdot n^3$

$C \geq 1$

→ Big $\cap = \cap(n^3)$

$n^3 \geq C \cdot n^3$

$C \leq 1$

→ Big $\theta = \theta(n^3)$

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

$\Rightarrow \theta(n^3)$ benar

4. for $i \leftarrow 1$ to n do

for $j \leftarrow i$ to n do

$M_{ij} \leftarrow a_{ij} + b_{ij}$

end for

end for

$T(n) = n^2$

→ big-O = $O(n^2)$

$n^2 \leq C \cdot n^2$

$C \geq 1$

→ big $\cap = \cap(n^2)$

$n^2 \geq C \cdot n^2$

$C \leq 1$

→ big $\theta = \theta(n^2)$

karena $O(n^2) = \cap(n^2), \theta(n^2) \checkmark$

5. for $i \leftarrow 1$ to n do
 $a_i \leftarrow b_i$
end for

$$T(n) = n$$

$$\rightarrow \text{big-O} = O(n)$$

$$n \leq c \cdot n$$

$$c \geq 1$$

$$\rightarrow \text{big-}\Omega = \Omega(n)$$

$$n \geq c \cdot n$$

$$c \geq 1$$

$$\rightarrow \Theta(n), \text{ karena } O(n) = \Omega(n)$$

$\Theta(n)$ benar

6. a). Operasi perbandingan

$$T(n) = (n-1) + (n-2) + (n-3) + \dots + 1$$

$$= \frac{n(n-1)}{2} = \frac{n^2 - n}{2}$$

b). Max pertukaran terjadi saat

$$\frac{n(n-1)}{2}$$

c). $\rightarrow T(n)$ saat best case

$$\frac{n(n-1)}{2} \text{ kali}$$

$$T(n) = \frac{n(n-1)}{2} = \frac{n^2 - n}{2}$$

$\rightarrow T(n)$ saat worst case

$$\text{Perbandingan} \rightarrow \frac{n(n-1)}{2}$$

$$\text{Assignment} \rightarrow \frac{3n(n-1)}{2}$$

$$T_{\max}(n) = \frac{n(n-1)}{2} + \frac{3n(n-1)}{2} = \frac{4n(n-1)}{2}$$

$$= 2n^2 - 2n$$

$$* O(n^2)$$

$$2n^2 - 2n \leq c \cdot n^2$$

$$2 - \frac{2}{n} \leq c$$

$$n=1 \Rightarrow 2-2 \leq c$$

$$c \geq 0$$

$$* \Omega(n^2)$$

$$\frac{n^2 - n}{2} \geq c \cdot n^2$$

$$\frac{1}{2} - \frac{1}{2n} \geq c$$

$$n=1 \Rightarrow \frac{1}{2} - \frac{1}{2} \geq c$$

$$c \leq 0$$

7. a). Algoritma A $\rightarrow O(\log n)$

b). Algoritma B $\rightarrow O(n \log n)$

c). Algoritma C $\rightarrow O(n^2)$

$$n = 8$$

$$A \rightarrow O(\log 8) = O(3 \log 2)$$

$$B \rightarrow O(8 \log 8) = O(21 \log 2)$$

$$C \rightarrow O(8^2) = O(64)$$

Algoritma tercepat = A

8. Operasi Assignment

$$\Rightarrow P_2$$

$$\rightarrow b_n \leftarrow a_n$$

$$\rightarrow b_k \leftarrow a_k + b_{k+1}$$

$$T(n) = 1 + n$$

$$O(n) = O(P_2)$$

$$\Rightarrow P$$

Pertambahan = n kali

Pertukaran = n kali

$$T(n) = 2n$$

P_2 lebih baik karena lebih kecil