Nama : Muhammad Luthfiansyah
NPM : 140810170023

Tugas 3 Analgo

1. Untuk $T(n) = 2+4+8+16+\cdots+2^n$, tentukan nilai : C, f(n), n_0, dan notasi Big-O

Jawaban :

T(n) adalah deret geometri

$$S(n) = \frac{a(r^n-1)}{r-1} = \frac{2(2^n-1)}{2-1} = 2.2^n - 2$$

$$T(n) = 2.2^n - 2 = 2^{n+1} - 2$$

$\boxed{T(n) = O(2^n) \rightarrow \text{notasi Big O}}$ dan $\boxed{f(n) = 2^n}$

Untuk $c=2$ dan semua $n \geq 0$ dan $n_0 = 0$ dapat dibuktikan $T(n) \leq c.f(n)$

$$2^{n+1} - 2 \leq c.2^n$$

$$\frac{2^{n+1}}{2^n} - \frac{2}{2^n} \leq c$$

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

$c \geq 0$ maka $\boxed{n_0 = 1 \text{ dan } c \geq 1}$

2. Buktikan bahwa untuk konstanta - konstanta positif p, q, dan r : $T(n) = pn^2 + qn + r$ adalah $O(n^2)$, $\Omega(n^2)$ dan $\Theta(n^2)$

Jawab: •) $pn^2 + qn + r \leq pn^2 + qn^2 + rn^2 \rightarrow T(n) \leq c.f(n) \rightarrow O(n^2)$

jika $n=1 \rightarrow p+q+r \leq p+q+r \Rightarrow$ terbukti

•) $T(n) \geq c.f(n)$

$pn^2 + qn + r \geq c.f(n)$

$pn^2 + qn + r \geq c.n^2$

$p + \frac{q}{n} + \frac{r}{n^2} \geq c$

$c = p+q+r$

$= 1+1+1 = 3 \rightarrow c \leq 3$ dan $n=1$

$T(n) = pn^2 + qn + r \rightarrow \Omega(n^2) \Rightarrow$ terbukti

•) $T(n) = O((f(n))$ Jika $T(n)$ adalah $O(f(n))$ dan $\Omega(f(n))$

$f(n) = n^2$, maka $T(n) = O(f(n)) = \Omega((f(n)) \rightarrow \Theta(n^2) \Rightarrow$ terbukti

3. Tentukan waktu kompleksitas asimtotik (Big-O, Big-Ω, dan Big-θ) dari program berikut

```
for k ← 1 to n do                          O(n)
    for i ← 1 to n do                      O(n)
        for j ← 1 to n do                  O(n)
            wij ← wij or wik and wkj       O(1)
        endfor
    endfor
endfor
```

Jawaban:

$T(n) = O(n) \cdot O(n) \cdot O(n) \cdot O(1) = O(n^3) \rightarrow f(n)$

Big-O $= O(f(n)) = O(n^3)$

Big Ω $= \Omega(f(n)) = O(n^3)$

Big θ → terpenuhi karena Big-O = Big-Ω

Big θ $= \theta[\in(n)] = \theta(n^3)$

4. Tulis Algoritma untuk menjumlahkan 2 matriks masing-masing berukuran nxn. Berapa kompleksitas waktunya T(n)? Berapa kompleksitas waktunya dalam asimtotik (O, Ω, θ)?

Jawab: Deklarasi i, j = integer

Algoritma

```
(1) for i ← 1 to n do
(2)     for j ← to n do
(3)         hasil [i,j] ← a [ij] + b [ij]
(4)     endfor
(5) endfor
```

maka $T(n) = n.n = n^2 \rightarrow f(n)$

T(n) sudah berlaku untuk worst case

sehingga Big-O = Big-Ω

Big-O $= O((f(n)) = O(n^2)$

Big-Ω $= \Omega((f(n)) = \Omega(n^2)$

Karena Big-O = Big-Ω maka Big-θ =

Big-O = Big-Ω = $\theta(f(n))$

5. Tulislah algoritma untuk mengcopy isi sebuah larik ke larik lain. Untuk ukuran elemen larik adalah n. Berapa kompleksitas waktunya? Berapa kompleksitas waktu Asimtotiknya?

Jawaban:

Algoritma

(1) for i ← 1 to n do → n kali / O(n)

(2) b[i-1] ← a[i-1]

(3) endfor

Maka $T(n) = n$

$T(n)$ = worst case & Best case juga

sehingga Big-O = Big Ω

$T(n) = n$, $O(f(n)) = O(n)$, $\Omega(f(n)) = \Omega(n)$ } $O(f(n)) = \Theta(n)$

6. Diberikan bubble sort

- Berapa jumlah operasi perbandingan elemen ≧ ?
- Berapa kali maks pertukaran elemen tabel?
- Hitung kompleksitas asimtotiknya

Jawaban:

Jumlah Operasi Perbandingan

Pass	Jumlah Operasi
1	n-1
2	n-2
3	n-3
:	:
n	1

maka :

$T(n) = (n-1) + (n-2) + \cdots + 1$

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

- Maksimum pertukaran elemen = $\frac{n(n-1)}{2}$

- Kompleksitas waktu Asimtotik

Big-O → $T(n) = \frac{n^2-n}{2} = O(n^2)$

Big-Ω → $T(n) = n^2+n = \Omega(n^2)$

Big-Θ = $T(n) = n^2+n = \Theta(n^2)$

7). Problem x dengan ~~merge~~ ukuran n = 8, algoritma mana yang paling cepat? Algoritma mana yang paling cepat? juga secara asimtotik?

- Algoritma A, $T(n) = O(\log N) = O(\log 8)$
- Algoritma B, $T(n) = O(N \log N) = O(8 \log 8)$
- Algoritma C, $T(n) = O(N^2) = O(64)$

Jawaban : semakin kecil bilangan di dalam kurung, semakin sedikit operasi yang dikerjakan

8). Algoritma mengevaluasi polinom yang lebih baik. dapat dibuat dengan metode Horner :

$P(x) = a_0 + x.(a_1 + x(a_2 + \dots + x(a_{n-1} + a_n x))$

Fungsi p^2 (input x : real) → real

Deklarasi

k = integer

$b_1, b_2, \dots, b_n$ = real

Algoritma

$b_n \leftarrow a_n$

for $k \leftarrow n-1$ down to 0 do

$b_x \leftarrow (a_x + b_{x+1})\ x$

endfor

return b_0

∴ Hitung operasi kali & jumlah nya, jumlahkan ke 2 nya dan tentukan KWA nya terbaik p atau p_2?

Jawaban :

- Algoritma p → Jumlah = n kali, kali = n kali $T(n) = n+n = 2n = n$
- Algoritma p2

  $T_2(n) = 1+n = O(n)$
- Kesimpulan : Sama-sama baik karena $T(n)$ & kompleksitas waktu Asimtotik nya sama bernilai $O(n)$