

LAPORAN PRAKTIKUM
ANALISIS ALGORITMA



DISUSUN OLEH

Gede Bagus Darmagita - 140810180068

PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PADJADJARAN
2020

Latihan Analisa

Minggu ini kegiatan praktikum difokuskan pada latihan menganalisa, sebagian besar tidak perlu menggunakan komputer dan mengkode program, gunakan pensil dan kertas untuk menjawab persoalan berikut!

1. Untuk $T(n) = 2 + 4 + 6 + 8 + 16 + \dots + n^2$, tentukan nilai C , $f(n)$, n_0 , dan notasi Big-O sedemikian sehingga $T(n) = O(f(n))$ jika $T(n) \leq C$ untuk semua $n \geq n_0$
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 $\Theta(n^2)$
3. Tentukan waktu kompleksitas asimptotik (Big-O, Big- Ω , dan Big- Θ) dari kode program berikut:


```

for k ← 1 to n do
  for i ← 1 to n do
    for j ← 1 to n do
       $w_{ij} \leftarrow w_{ij}$  or  $w_{ik}$  and  $w_{kj}$ 
    endfor
  endfor
endfor
      
```
4. Tulislah algoritma untuk menjumlahkan dua buah matriks yang masing-masing berukuran $n \times n$. Berapa kompleksitas waktunya $T(n)$? dan berapa kompleksitas waktu asimptotiknya yang dinyatakan dalam Big-O, Big- Ω , dan Big- Θ ?
5. Tulislah algoritma untuk menyalin (copy) isi sebuah larik ke larik lain. Ukuran elemen larik adalah n elemen. Berapa kompleksitas waktunya $T(n)$? dan berapa kompleksitas waktu asimptotiknya yang dinyatakan dalam Big-O, Big- Ω , dan Big- Θ ?

6. Diberikan algoritma Bubble Sort sebagai berikut:

```

procedure BubbleSort(input/output  $a_1, a_2, \dots, a_n$ ; integer)
  ( Mengurut tabel integer TabInt[1..n] dengan metode pengurutan bubble-
  sort
  Masukan:  $a_1, a_2, \dots, a_n$ 
  Keluaran:  $a_1, a_2, \dots, a_n$  (terurut menaik)
  )
  Deklarasi
    k : integer ( indeks untuk traversal tabel )
    pass : integer ( tahapan pengurutan )
    temp : integer ( peubah bantu untuk pertukaran elemen tabel )
  Algoritma
    for pass  $\leftarrow 1$  to  $n - 1$  do
      for k  $\leftarrow n$  downto pass + 1 do
        if  $a_k < a_{k-1}$  then
          ( pertukarkan  $a_k$  dengan  $a_{k-1}$  )
          temp  $\leftarrow a_k$ 
           $a_k \leftarrow a_{k-1}$ 
           $a_{k-1} \leftarrow temp$ 
        endif
      endfor
    endfor
  endfor

```

- Hitung berapa jumlah operasi perbandingan elemen-elemen tabel!
- Berapa kali maksimum pertukaran elemen-elemen tabel dilakukan?
- Hitung kompleksitas waktu asimptotik (Big-O, Big- Ω , dan Big- Θ) dari algoritma Bubble Sort tersebut!

7. Untuk menyelesaikan problem X dengan ukuran N tersedia 3 macam algoritma:

- Algoritma A mempunyai kompleksitas waktu $O(\log N)$
- Algoritma B mempunyai kompleksitas waktu $O(N \log N)$
- Algoritma C mempunyai kompleksitas waktu $O(N^2)$

Untuk problem X dengan ukuran $N=8$, algoritma manakah yang paling cepat? Secara asimptotik, algoritma manakah yang paling cepat?

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

$$p(x) = a_0 + x(a_1 + x(a_2 + x(a_3 + \dots + x(a_{n-1} + a_n x))) \dots))$$

```

function p2(input x : real)  $\rightarrow$  real
  ( Mengembalikan nilai  $p(x)$  dengan metode Horner )

```

Deklarasi

```

  k : integer
   $b_1, b_2, \dots, b_n$  : real

```

Algoritma

```

   $b_n \leftarrow a_n$ 
  for k  $\leftarrow n - 1$  downto 0 do
     $b_k \leftarrow a_k + b_{k+1} * x$ 
  endfor
  return  $b_0$ 

```

Hitunglah berapa operasi perkalian dan penjumlahan yang dilakukan oleh algoritma diatas, Jumlahkan kedua hitungan tersebut, lalu tentukan kompleksitas waktu asimptotik (Big-O)nya. Manakah yang terbaik, algoritma p atau p2?

Jawaban:

Gede Bagus D

Modul 10 - B

Tugas 3 - Analisis

1. $T(n)$ = deret geometri

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

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

$$T(n) = O(2^n) \text{ dan } f(n) = 2^n$$

$$\Rightarrow T(n) \in C \cdot f(n)$$

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

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

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

$$2 - 1 \leq C$$

$$1 \leq C \rightarrow \boxed{C=1, C \geq 1}$$

$$2. Pn^2 + an + r = O(n^2)$$

$$Pn^2 + an + r \leq n^2 \leq n^2$$

$$P + \frac{a}{n} + \frac{r}{n^2} \leq C ; n=1$$

$$P + a + r \leq C$$

terbukti bahwa $P_0=1$ dan $C \geq P+a+r$

$$Pn^2 + an + r = \Omega(n^2)$$

$$Pn^2 + an + r \geq n^2 : n^2$$

$$P + \frac{a}{n} + \frac{r}{n^2} \geq C ; n=1$$

$$P + a + r \geq C$$

terbukti bahwa $P_0=1$ dan $C \leq P+a+r$

\Rightarrow Karena $\text{big } O = \text{big } \Omega = n^2$

maka $\text{big } \Theta = n^2$

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

$$- \text{big } O = O(f(n)) = O(n^3)$$

$$- \text{big } \Omega = \Omega(n^3)$$

$$- \text{big } \Theta = \Theta(n^3), \text{ karena big } O \text{ dan big } \Omega \text{ sama.}$$

$$4. \text{ For } i \leftarrow 1 \text{ to } n \text{ do } O(n) \\ \text{ for } j \leftarrow 1 \text{ to } n \text{ do } O(n)$$

 make $[i,j] \leftarrow a[i,j] + b[i,j]$
 endfor
 endfor

$$T(n) = O(n) + O(n) \\ = O(n^2) \rightarrow f(n)$$

$$- \text{big } O = O(n^2)$$

$$- \text{big } \Omega = \Omega(n^2)$$

$$- \text{big } \Theta = \Theta(n^2)$$

$$5. \text{ For } i \leftarrow 1 \text{ to } n \text{ do } O(n) \\ \quad b[i-1] \leftarrow a[i-1]$$

endfor

$$T(n) = O(n) \rightarrow f(n)$$

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

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

$$- \text{big } \Theta = \Theta(n)$$

6a. Pars 1 jumlah operasi

$$\begin{array}{c} 1 \\ 2 \\ 3 \\ \vdots \\ n \end{array} \quad \begin{array}{c} n-1 \\ n-2 \\ n-3 \\ \vdots \\ 1 \end{array}$$

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

6b. Maka Pertanyaan elemen adalah $\frac{n(n-1)}{2}$

6c. Kompleksitas waktu asimptotik

- 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. algoritma A = $O(\log \theta)$

u B = $O(8 \log \theta)$

u C = $O(6u)$

Yang tercepat adalah algoritma A
karena memiliki $f(n)$ terkecil.

8. - Algoritma $p \rightarrow$ jumlah = n hari
 hari = n hari
 $T(n) = n + n = 2n = n \rightarrow O(n)$

- Algoritma $p_2 = T(n) = 1 + n$
 $= O(n)$

- p dan p_2 sama sama bagus.
 karena memiliki big O yang
 sama.