

Ahina Vania Kirana

140810180010 - KIS B

Tugas Praktikum Alprog.

① $T(n) = 2 + 4 + 6 + 8 + 16 + \dots + n^2$

Tentukan nilai $c, f(n)$, notasi Big-O!

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

$2 + 4 + 6 + 8 + 16 + \dots + n^2 = O(f(n))$

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

$2(2^n - 1) = O(f(n)) \rightarrow (2^n - 2) = O(f(n))$

$T(n) \leq C$

$T(n) = O(2^n)$

$2^n - 2 \leq C(2^n)$

$2 - 1 \leq C$

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

② $T(n) = pn^2 + qn + r$, Buktikan p, q, r positif!

$O(n^2), \Omega(n^2), \Theta(n^2)$

Big O

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

$pn^2 + qn + r \leq C(n^2)$

$p + \frac{q}{n} + \frac{r}{n^2} \leq C \rightarrow n \geq 1$

$p + q + r \leq C$ ④

Big Ω

$T(n) \geq f(n)$

$pn^2 + qn + r \geq C(n)$

$pn + q + \frac{r}{n} \geq C \rightarrow n \geq 1$

$p + q + r \geq C$ ④

Big Θ sama ordennya karena $\Theta(n) = \Omega(n)$

③ Tentukan kompleksitas asimtotik (Big-O, Big Ω , Big Θ) dari kode berikut.

for $k \leftarrow 1$ to n do

for $i \leftarrow 1$ to n do

for $j \leftarrow 1$ to n do

Wij \leftarrow Wij or Wik

and Wkj

end for

end for

end for

Big Θ

Big $\Omega =$ Big O

maka Big Θ sama.

Big O

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

Big Ω

$T(n) \geq f(n)$

$n^3 \geq C(n^3)$

$1 \geq C$

$1 \geq C$

④ Tulis Algoritma Matriks $n \times n$, tentukan kompleksitas waktu dan kompleksitas waktu asimtotik dan Big O, Ω , Θ !

algoritma

for $i \leftarrow 1$ to n do

for $j \leftarrow 1$ to n do

$mij \leftarrow aij + bij$

end for

end for

Big O = Big Ω

Maka Big Θ

Suma hasilnya.

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

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

$1 \leq C$

$T(n) \geq f(n)$

$n^2 \geq C(n^2)$

⑤ Tulis Algoritma menyalin, ukuran elemen bank adalah n elemen. $T(n)$? Big O, Ω , Θ ?

algoritma

for $i \leftarrow 1$ to n do

$a_i \leftarrow b_i$

end for

Big Θ

Big $\Omega =$ Big Ω

maka Big Θ

Juga sama.

Big Ω

$T(n) \geq f(n)$

$n \geq C(n)$

$1 \geq C$

Big O

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

$n \leq C(n)$

$1 \leq C$

- Hitung jumlah operasi perbandingan elemen: tabel
- Berapa kali max pertukaran elemen: tabel dibalikkan
- Hitung komp. waktu asimtotik Big O, Ω , Θ !

algoritma bubble sort

for pass $\leftarrow 1$ to $(n-1)$ do

for $k \leftarrow n$ down to pass+1 do

if $a_k < a_{k-1}$ then

$a_k \leftrightarrow a_{k-1}$

temp $\leftarrow a_k$

$a_k \leftarrow a_{k-1}$

$a_{k-1} \leftarrow temp$

end if

end for

end for

Big O

$2n^2 - 2n \leq Cn^2$

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

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

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

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

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

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

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

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

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

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

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

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

Best case
Worst case
assignment

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

Operasi perbandingan
didlm loop.

$1 \rightarrow (n-1)$

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

7) Selesaikan problem x dgn ukuran N , tersedia 3 algoritma :

- a. Algoritma A $\rightarrow O(\log N)$
- b. Algoritma B $\rightarrow O(N \log N)$
- c. Algoritma C $\rightarrow O(N^3)$

Jika $N=8$ algoritma manakah yg paling cepat

- a. $O(\log 8) : O(3 \log_2)$
- b. $O(8 \log_2 8) : O(8^3 \log_2 8) : O(2^4 \log_2 8)$
- c. $O(8^3) : O(64)$

Yang paling efektif algoritma A krn paling kecil nilainya.

8) algoritma

```

bn ← an
for k ← n-1 down to 0 do
    bk ← ak + bk+1 * x
endfor
return b0

```

hitunglah berapa operasi
perkalian dan penjumlahan
yang dilakukan oleh algoritma
manakah yang lebih
baik p / p₂

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

P
Jumlah : n kali
kali : n kali

P₂
P₂(n) : n+1

T(0) : n
= O(n)

$$T(n) = 2n \\ = O(n)$$

Sama, keduanya big O nya sama yaitu n.

