Nama : Fadhli Hibatul Haqqi

NPM : 140810170019

Kelas : A

1.

(a)

1 for j = 1 to n-1 //n-1 looping

2 k = j //n-1 operasi

3 for i = j + 1 to n //

4 if a[i] < a[k] then //

5 k = i //

6 endif

7 endfor

8 tm = a[j] //n-1 operasi

9 a[j] = a[k] //n-1 operasi

10 a[k] = tm //n-1 operasi

11 endfor

Pada loop baris ke 3 akan lopping sebanyak

|  |  |  |
| --- | --- | --- |
| j | j + 1...n | Loop |
| 1 | 2...n | n-1 |
| 2 | 3...n | n-2 |
| ... | ... | ... |
| n-2 | n-1...n | 2 |
| n-1 | n...n | 1 |

1 + 2 + ... + (n-2) + (n-1)

S(n-1) =

T(n) = 2() + 4 (n-1) = = = O()  
T(n) ≤ C f(n)

≤ C

≤ C

N0 = 1

1 + 3 – 4 ≤ C

C ≥ 0

(b)

1 for i = 0 to n-1 //n looping

2 for j = 0 to n-1 // n \* n = looping

3 c[i,j] = 0 // operasi

4 for k = 0 to n-1 // looping

5 cij = d[i,k] and b[k,j] // operasi

6 c[i,j] = c[i,j] or cij  // operasi

7 endfor

8 endfor

9 endfor

T(n) = 2() + = = O()

T(n) ≤ C f(n)

≤ C

≤ C

N0 = 1

2 + 1 ≤ C

C ≥ 3

2.

(a)

1 ada = 0; // 1 operasi

2 kx = 1; // 1 operasi

3 input br;

4 for (i = 1; i < n + 1; i++) // n looping

5 { if(a[i] == br && (!ada)) // n operasi

6 { ada = 1; // 1 operasi

7 kx = i; // 1 operasi

8 i = n+1; // 1 operasi

9 }

10 }

T(n) = = O(n)

T(n) ≤ C f(n)

≤ C

≤ C

N0 = 1

1 + 5 ≤ C

C ≥ 6

(b)

1 L = 1; // 1 operasi

2 R = n; // 1 operasi

3 ada = 0; // 1 operasi

4 input = br;

5 while((L <= R) && (!ada)) // looping

6 { m = (L + R) div 2; // operasi

7 if(a[m] == br) // operasi

8 ada = 1; // operasi

9 else if (br < a[m]) // baris 9 – akhir 2()

10 R = m – 2;

11 else

12 L = m + 1;

13 }

Baris 5 akan looping sebanyak :

|  |  |
| --- | --- |
| k | n |
| 1 |  |
| 2 |  |
| 3 |  |
| ... | ... |
| k |  |

Untuk worst case n akan dibagi sampai menyisakan 1 elemen maka,

Sehingga , dan dapat diubah ke bentuk logaritma,

T(n) = 2 () + 2( - 1) + 4 =

T(n) ≤ C f(n)

≤ C

≤ C

N0 = 2

4 + 2 ≤ C

C ≥ 6

Bagian B

Komputer A mengeksekusi instruksi/detik

Jumlah data

Kompleksitas waktu = O(n)

detik

Komputer B mengeksekusi instruksi/detik

Jumlah data

Kompleksitas waktu = O()

= 8 \* 3,32 \* 10-7 detik = 26,56 \* 10-7 detik = 0,027 \* 10-4 detik.

Jadi, Algoritma 2 b lebih baik karena dapat di running lebih cepat dari algoritma 2 a walaupun komputer B lebih lambat dalam mengeksekusi instruksi