Nama : Rizal Fathur Rahman

NPM : 140810170009

Praktikum Analgo

1. Soal 1. a.

For j = 1 to n-1   
 k = j \\ n-1

For i = j + 1 to n   
 if a[i] < a[k] then  
 k=i \\ 2 ∑ [ i=2 sampai n](n/2 (2+n)) = 2n + n2  
 endif  
 endfor  
 tm=a[j] \\ operasi assignment  
 a[j] = a[k] \\ operasi assignment  
 a[k] = tm \\ operasi assignment  
endfor

T(n) = (n-1) + 2n+n2 + (n-1) + (n-1) + (n-1)  
 = 6n + n2 - 4 = O (n2)  
  
Kompleksitas O  
T(n) <= c . f(n)  
n2 + 6n – 4 <= c . n21 + 6/n – 4/n2 <= c  
Misal n0 = 1  
1 + 6 – 4 <= c  
3 <= c  
c >= 3

Big O terbukti positif dengan n0 = 1 , c >= 3 maka Big O = O(n2)  
  
1.b.   
for i=0 to n-1  
 for j=0 to n-1  
 c[i,j] = 0 \\ n-1  
 for k=0 to n-1  
 cij= d[i,k] and b[k,j] \\ operasi assignment  
 c[i,j] = c[i,j] or cij \\ operasi assignment  
 endfor  
 endfor  
endfor

T(n) = (n-1)(n-1)(n-1)   
 = (n2 -2n + 1)(n-1)= (n3 – 3n2 + 3n – 1)  
 T(n) = O(n3)

Kompleksitas O  
T(n) <= c.f(n)  
(n3 – 3n2 + 3n – 1) <= c. n31 – 3/n + 3/n2 – 1/n3) <= c  
misal n0 = 2  
1 – 3 + 3 – 1 <= c

c >= 0,125  
  
Big O terbukti positif dengan n0 = 2 , c >= 0,125 maka Big O = O(n3)

1. Soal no 2a.  
   a) Ada = 0; \\ n

Kx=1; \\ n

Input br;

For (i=1; i<n; i++){

If(a[i] == br && (!ada)){

Ada = 1; \\ n

kx = 1; \\ n

i = n+1; \\ ∑ k [ i=1 sampai n](n/2 (n+1))

}

}

T(n) = n/2(n+1)  
T(n) = O (n)  
  
Kompleksitas O  
T(n) <= c. f(n)  
n/2(n+1) <= c. n

½(n+1) <= c  
misal n0 = 1  
½ (1+1) <= c  
c >= 1

Big O terbukti positif dengan n0 = 1 , c >= 1 maka Big O = O(n)

Soal no 2b.  
L = 1; \\ 1

R=n; \\ 1

Ada = 0; \\ 1

Input br;

While((L<= R) && (!ada)){

M = (L+R) div 2;

If (a[m] == br)

Ada = 1;

Else if (br<a[m])

R = m+1;

Else

L=m+1;

}

Dari m = (L+R) div 2 didapat   
loop 1 n/2   
loop 2 n/2  
loop k n/2k

n/2k = 1

n = 2k

k =   
  
oleh karena itu   
M = (L+R) div 2; \\

If (a[m] == br) \\

Ada = 1; \\ 1

Else if (br<a[m]) \\

R = m+1 atau L=m+1 \\

T(n) = 1 + 1 + 1 + + + 1 + +   
 = 4 + 4/   
 T(n) = O()

Kompleksitas O   
 T(n) <= c. f(n)  
4 + 4( ) <= c.

4 + 4 / <= c  
misal n0 = 2  
4 + 4 <= c

8 <= c  
c >= 8  
  
Big O terbukti positif dengan n0 = 2, c >= 8 maka Big O = O()

Menghitung running time :

Running time = jumlah instruksi / kecepatan eksekusi  
Running time algoritma a :  
108/109 = 10-1 detik

Running time algoritma b:

(8)/107 = 26.57542 x 10-7 = 2,657542 x 10-8 detik-

Algoritma yang lebih baik adalah algoritma yang memiliki efisiensi waktu lebih baik, yaitu algoritma b