

## Tugas 1 - Kompleksitas Algoritma.

Nama Kelompok : No Name

Mata Kuliah : Analisis Algoritma

Semester Ganjil 2019/2020



### Pernyataan anggota kelompok.

Saya yang bertanda tangan dibawah ini menyatakan berkontribusi (mengerjakan dan mampu memahami dan menjelaskan bagian pekerjaan saya) dalam mengerjakan tugas ini. Jika nanti terbukti berbohong, saya siap dikeluarkan dari kelompok ini dan semua kontribusi yang pernah saya berikan di kelompok ini dibatalkan.

Mataram, 30 Oktober 2019

*Hand*

Almas Ishtahat W.  
NIM: 1710510066

*Elf*

Mugilud ali ST  
NIM: 1710510084

*Dunt*

Alga Deni S.  
NIM: 1710510064

*HB*

Nining Sumatra  
NIM: 1710510083

*Part*

Andi Sultansyah FR  
NIM: 1710510074

Tugas 1 - Kompleksitas Algoritma  
Nama Kelompok : No Name  
Mata Kuliah Analisis Algoritma  
Semester Ganjil 2019/2020

Anggota Kelompok :

Alimad Ikhul Wathan (1710510066)  
Muzahid Ali Gharizah T (1710510084)  
Alga Doni Saputra (1710510064)  
Nanang Samudra (1710510089)  
Andi Sultansyah Fadila Rahmat (1710510074)

Universitas Bumi Gora  
Jl. Ismail Marzuki, Motaram.



Bubble Sort

int Bubble (int A[])

for (int i = 0; i < A.length - 1; i++)

Art	Asg	C
$n-2$	1	$n-2$
$(n-2)^2$	$n-2$	$(n-1) - 2$
—	—	✓
—	x	—
—	x	—
—	x	—

$$\begin{aligned}
 & \Rightarrow n^2 - 3n + 2 + n - 3 + 2x + 2n - 3 \\
 & \Rightarrow \frac{n-2}{2} (n-2)^2 \quad n-2 (n-2)^2 \\
 & \Rightarrow \frac{n-2}{2} + n^2 - 4n + 4 \quad n-2 + n^2 - 4n + 4 \\
 & \Rightarrow n^2 - 3n + 2 \\
 & \Rightarrow n^2 + 6n - 4 + 3x
 \end{aligned}$$

Worst case dengan  $x = n$

$$\begin{aligned}
 T(n) &= 6n + 21 + 8n \\
 &= 14n + 21
 \end{aligned}$$

$$f(n) = 14n + 21$$

$$fn = 14n + 21$$

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

$$14n + 21 \leq 14n + 21n$$

$$f(n) = O(n) \text{ dengan } C = 35 \text{ dan } n_0 = 1$$

$$\begin{aligned}
 T(n) &= (T(n/2) + T(n/2)) + O(n) \\
 &= 2T(n/2) + O(n)
 \end{aligned}$$

$$T(n) = 2^{(n \log n)}$$



Merge Sort

Data  $\rightarrow (1, \dots, n)$

void merge (int arr[], int l, int m, int r &  
int i, j, k  
int n1 = ~~m~~ m - l + 1,  
int n2 = r - m

in L[n1] = arr[l]

for (i = 0, j = l, i < n1, i++) &  
L[i] = arr[l + i]

for (j = 0, ~~j = m~~ j = m + 1, j < n2, j++) &  
~~R[j] = arr[m + 1 + j]~~ R[j] = arr[m + 1 + j]

int = i, j = 0

int x = 1;

while (i < n1 & ~~x + j~~ x + j < n2) &

if (L[i] < R[j]) &

arr[x] = L[i];

i++

}

else &

arr[x] = R[j]

j++

}

}

x++ }