

LAPORAN PRAKTIKUM 4
ANALISIS ALGORITMA



Disusun oleh :

Nurul Ma'arif

140810180040

PROGRAM STUDI S-1 TEKNIK INFORMATIKA
FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS PADJADJARAN
2020

Studi Kasus 1: Merge Sort

1. Source code

```
#include <iostream>
using namespace std;

void merge(int *,int, int , int );
void merge_sort(int *arr, int low, int high)
{
    int mid;
    if (low < high){
        mid=(low+high)/2;
        merge_sort(arr,low,mid);
        merge_sort(arr,mid+1,high);

        merge(arr,low,high,mid);
    }
}

void merge(int *arr, int low, int high, int mid)
{
    int i, j, k, c[50];
    i = low;
    k = low;
    j = mid + 1;
    while (i <= mid && j <= high) {
        if (arr[i] < arr[j]) {
            c[k] = arr[i];
            k++;
            i++;
        }
        else {
            c[k] = arr[j];
            k++;
            j++;
        }
    }
    while (i <= mid) {
        c[k] = arr[i];
        k++;
        i++;
    }
    while (j <= high) {
        c[k] = arr[j];
        k++;
        j++;
    }
    for (i = low; i < k; i++) {
        arr[i] = c[i];
    }
}

int main()
{
    int myarray[30], num;

    cout<<"Masukkan jumlah elemen: ";
    cin>>num;

    for (int i = 0; i < num; i++) {
```

```

        cout << "Elemen " << i+1 << " : ";
        cin>>myarray[i];
    }

    merge_sort(myarray, 0, num-1);
    cout<<"\nSorted array\n";

    for (int i = 0; i < num; i++){
        cout<<myarray[i]<<" ";
    }
}

```

```

C:\Users\asus\Documents\College\Sem 4\prak analgo\AnalgoKu\AnalgoKu4
Masukkan jumlah elemen: 20
Elemen 1 : 99
Elemen 2 : 12
Elemen 3 : 13
Elemen 4 : 21
Elemen 5 : 88
Elemen 6 : 77
Elemen 7 : 25
Elemen 8 : 33
Elemen 9 : 85
Elemen 10 : 86
Elemen 11 : 1
Elemen 12 : 5
Elemen 13 : 2
Elemen 14 : 6
Elemen 15 : 9
Elemen 16 : 10
Elemen 17 : 100
Elemen 18 : 66
Elemen 19 : 46
Elemen 20 : 77

Sorted array
1 2 5 6 9 10 12 13 21 25 33 46 66 77 77 85 86 88 99 100
Process returned 0 (0x0)   execution time : 47.924 s
Press any key to continue.

```

2. Kompleksitas waktu merge sort = $O(n \lg n) \rightarrow T(20 \log_{10} 20)$

Studi Kasus 2: Selection Sort

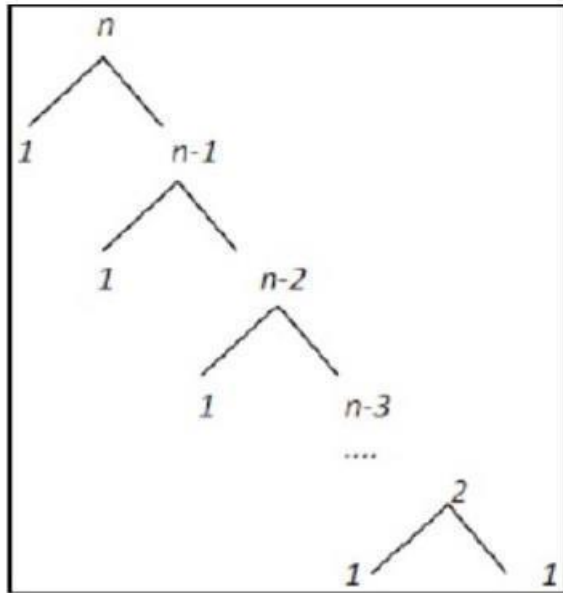
$T(n)$ selection sort

```

for i ← n downto 2 do {pass sebanyak n-1 kali} imaks ← 1
    for j ← 2 to i do
        if  $x_j > x_{imaks}$  then
            imaks ← j
        endif
    endfor
    {pertukaran  $x_{imaks}$  dengan  $x_i$ }
    temp ←  $x_i$ 
     $x_i$  ←  $x_{imaks}$ 
     $x_{imaks}$  ← temp
endfor

```

Waktu proses penggabungan = $n \cdot (1 - (-1) + (-1))$



$$\begin{aligned} &= c((n-1)(n-2)/2) + cn \\ &= c((n^2-3n+2)/2) + cn \\ &= c(n^2/2) - (3n/2) + 1 + cn \\ &= \Omega(n^2) \end{aligned}$$

$$T(n) = cn^2$$
$$= \Theta(n^2)$$

```
#include<iostream>
using namespace std;

void swapping(int &a, int &b) {
    int temp;
    temp = a;
    a = b;
    b = temp;
}

void display(int *array, int size) {
    for(int i = 0; i<size; i++)
        cout << array[i] << " ";
}
```

```

        cout << endl;
    }

void selectionSort(int *array, int size)
{
    int i, j, imin;
    for(i = 0; i<size-1; i++) {
        imin = i;
        for(j = i+1; j<size; j++)
            if(array[j] < array[imin])
                imin = j;
        swap(array[i], array[imin]);
    }
}

int main() {
    int n;
    cout << "Enter the number of elements: ";
    cin >> n;

    int arr[n];
    cout << "Enter elements:" << endl;
    for(int i = 0; i<n; i++) {
        cin >> arr[i];
    }

    cout << "Array before Sorting: ";
    display(arr, n);
    selectionSort(arr, n);
    cout << "Array after Sorting: ";
    display(arr, n);
}

```

```

C:\Users\asus\Documents\College\Sem 4\prak analgo\AnalgoKu\Analog
Masukkan jumlah elemen: 10
Enter elements:
13
41
55
1
99
55
32
86
69
10
Array after Sorting: 1 10 13 32 41 55 55 69 86 99

Process returned 0 (0x0)   execution time : 19.419 s
Press any key to continue.

```

Studi Kasus 3: Insertion Sort

Algoritma:

```

for i ← 2 to n do
    insert ← xi
    j ← i
    while (j > 1 and (x[j-1] > insert)) do
        x[j] ← x[j-1]
        j ← j-1
    endwhile
    x[j] = insert
endfor

```

Subproblem = 1

Masalah setiap subproblem = n-1

Waktu proses pembagian = n

Waktu proses penggabungan = n

$T(n) = \{ (1) (n-1) + (n) \}$

$T(n) = cn + cn - c + cn - 2c + \dots + 2c + cn \leq 2cn + cn$

$= c((n-1)(n-2)/2) + cn \leq 2cn + cn$

$= c((n^2 - 3n + 2)/2) + cn \leq 2cn + cn$

$= c(n^2/2) - c(3n/2) + c + cn \leq 2cn +$

$cn = O(n^2)$

$T(n) = cn \leq cn$

$= \Omega(n)$

$T(n) = (cn + cn)/n$

$= \Theta(n)$

Source code

```
#include<iostream>
using namespace std;

void display(int *array, int size) {
    for(int i = 0; i<size; i++)
        cout << array[i] << " ";
    cout << endl;
}

void insertionSort(int *array, int size)
{ int key, j;
  for(int i = 1; i<size; i++) {
    key = array[i];
    j = i;
    while(j > 0 && array[j-1]>key) {
        array[j] = array[j-1];
        j--;
    }
    array[j] = key;
  }
}

int main() {
    int n;
    cout << "Enter the number of elements: ";
    cin >> n;

    int arr[n];
    cout << "Enter elements:" << endl;

    for(int i = 0; i<n; i++) {
        cin >> arr[i];
    }

    cout << "Array before Sorting: ";
    display(arr, n);
    insertionSort(arr, n);
    cout << "Array after Sorting: ";
    display(arr, n);
}
```

```

C:\Users\asus\Documents\College\Sem 4\prak analgo\AnalgoKu\Ana
Enter the number of elements: 7
Enter elements:
42
1
69
86
88
24
52
Array before Sorting: 42 1 69 86 88 24 52
Array after Sorting: 1 24 42 52 69 86 88

Process returned 0 (0x0)   execution time : 19.337 s
Press any key to continue.

```

Studi Kasus 4: Bubble Sort

Subproblem = 1

Masalah setiap subproblem = $n-1$

Waktu proses pembagian = n

Waktu proses penggabungan

$$T(n) = \{ (1) (n-1) + (n) \}$$

$$T(n) = cn + cn - c + cn - 2c + \dots + 2c + c \leq 2cn^2 + cn^2$$

$$= c((n-1)(n-2)/2) + c \leq 2cn^2 + cn^2$$

$$= c((n^2 - 3n + 2)/2) + c \leq 2cn^2 + cn^2$$

$$= c(n^2/2) - c(3n/2) + 2c \leq 2cn^2 + cn^2$$

$$= O(n^2)$$

$$T(n) = cn + cn - c + cn - 2c + \dots + 2c + c \leq 2cn^2 + cn^2$$

$$= c((n-1)(n-2)/2) + c \leq 2cn^2 + cn^2$$

$$= c((n^2 - 3n + 2)/2) + c \leq 2cn^2 + cn^2$$

$$= c(n^2/2) - c(3n/2) + 2c \leq 2cn^2 + cn^2$$

$$= \Omega(n^2)$$

$$T(n) = cn^2 + cn^2$$

$$= \Theta(n^2)$$

Source code

```

#include<iostream>

using namespace std;

int main()
{
    int a[50],n,i,j,temp;

```



```

    cout<<"Enter the size of array: ";
    cin>>n;

    for(i=0;i<n;++i){
        cout << "Elemen " << i+1 << " : ";
        cin>>a[i];
    }

    for(i=1;i<n;++i)
    {
        for(j=0;j<(n-i);++j)
            if(a[j]>a[j+1])
            {
                temp=a[j];
                a[j]=a[j+1];
                a[j+1]=temp;
            }
    }

    cout<<"Array after bubble sort:";
    for(i=0;i<n;++i)
        cout<<" "<<a[i];

    return 0;
}

```

```

C:\Users\asus\Documents\College\Sem 4\prak analgo\AnalgoKu\An
Enter the size of array: 5
Elemen 1 : 7
Elemen 2 : 1
Elemen 3 : 99
Elemen 4 : 100
Elemen 5 : 69
Array after bubble sort: 1 7 69 99 100
Process returned 0 (0x0)   execution time : 17.048 s
Press any key to continue.

```