

Algoritma & Struktur Data

## **10. Selection dan insertion sort**

**Dosen Pengampu**

Dr. Tita Karlita S.Kom, M.Kom



**Disusun Oleh :**

Nama : M. Faza Nur Husain

Nrp : 3121550004

**D3 PJJ AK TEKNIK INFORMATIKA  
POLITEKNIK ELEKTRONIKA NEGERI SURABAYA  
TAHUN AKADEMIK 2021/2022**

## Praktikum 10.1

### SORTING (INSERTION & SELECTION)

1. Buatlah program yang mengimplementasikan algoritma **selection** sort, untuk mengurutkan 10 buah **elemen** array bertipe integer.  
Tampilkan hasil pengurutan data dalam urutan ascending.  
Tampilkan **jumlah perbandingan** dan **jumlah penukaran elemen**.  
Gunakan data berikut ini dalam 4 kali running program untuk melihat efek dari distribusi elemen terhadap jumlah penukaran dan jumlah perbandingan.

array 1: 3, 10, 4, 6, 8, 9, 7, 2, 1, 5	→ data awal, semua elemen terdistribusi acak
array 2: 1, 2, 3, 4, 5, 9, 7, 6, 8, 10	→ data awal, 5 elemen pertama terurut descending, dan 5 elemen terakhir acak
array 3: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10	→ data awal, semua elemen terurut ascending
array 4: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1	→ data awal, semua elemen terurut descending

2. Buatlah program yang mengimplementasikan algoritma **insertion** sort, untuk mengurutkan 10 buah **elemen** array bertipe integer.  
Tampilkan hasil pengurutan data dalam urutan ascending.  
Tampilkan jumlah **perbandingan**, jumlah **pergeseran**, dan jumlah **penyisipan** elemen.  
Gunakan data berikut ini dalam 4 kali running program untuk melihat efek dari distribusi elemen terhadap jumlah penukaran dan jumlah perbandingan.

array 1: 3, 10, 4, 6, 8, 9, 7, 2, 1, 5	→ data awal, semua elemen terdistribusi acak
array 2: 1, 2, 3, 4, 5, 9, 7, 6, 8, 10	→ data awal, 5 elemen pertama terurut descending, dan 5 elemen terakhir acak
array 3: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10	→ data awal, semua elemen terurut ascending
array 4: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1	→ data awal, semua elemen terurut descending

## 1. Selection Sort

array 1: 3, 10, 4, 6, 8, 9, 7, 2, 1, 5	→ data awal, semua elemen terdistribusi acak
array 2: 1, 2, 3, 4, 5, 9, 7, 6, 8, 10	→ data awal, 5 elemen pertama terurut descending, dan 5 elemen terakhir acak
array 3: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10	→ data awal, semua elemen terurut ascending
array 4: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1	→ data awal, semua elemen terurut descending

Source Code :

```
#include <stdio.h>

int main()
{
    // Inisialisasi array
    int arr[] = {3, 10, 4, 6, 8, 9, 7, 2, 1, 5};
    int temp = 0;
    // Hitung panjang array arr
    int length = sizeof(arr) / sizeof(arr[0]);
    // Menampilkan elemen array asli
    printf("Semua elemen terdistribusi acak : \n");
    for (int i = 0; i < length; i++)
    {
        printf("%d ", arr[i]);
    }
    // ascending order setelah 5 random
    for (int i = 0; i < length - 5; i++)
    {
        for (int j = i + 1; j < length; j++)
        {
            if (arr[i] > arr[j])
            {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    printf("\n");
    // Menampilkan elemen array setelah menyortir
    printf("5 elemen pertama terurut Ascending, dan 5 elemen terakhir acak: \n");
    for (int i = 0; i < length; i++)
    {
        printf("%d ", arr[i]);
    }
    // ascending order
    for (int i = 0; i < length; i++)
    {
        for (int j = i + 1; j < length; j++)
        {
            if (arr[i] > arr[j])
            {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    printf("\n");
    // Menampilkan elemen array setelah menyortir
    printf("Semua elemen terurut ascending: \n");
    for (int i = 0; i < length; i++)
    {
```

```

        printf("%d ", arr[i]);
    }
    // descending order
    for (int i = 0; i < length; i++)
    {
        for (int j = i + 1; j < length; j++)
        {
            if (arr[i] < arr[j])
            {
                temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }
    }
    printf("\n");
    // Menampilkan elemen array setelah menyortir
    printf("Semua elemen terurut Descending: \n");
    for (int i = 0; i < length; i++)
    {
        printf("%d ", arr[i]);
    }
    return 0;
}

```

Output :

```

selectio-sort.c - Algoritma & Struktur Data - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS G:\My Drive\COLLAGE\FENS\SEMESTER 2\Algoritma & Struktur Data> cd "g:\My Drive\COLLAGE\FENS\SEMESTER 2\Algoritma & Struktur
Data\M10-Sorting 1 (insertion &selection)\tugas\praktikum\" ; if ($?) { gcc selectio-sort.c -o selectio-sort } ; if ($?) { .\s
electio-sort }

SELECTION SORT
-----

Semua elemen terdistribusi acak :
3 10 4 6 8 9 7 2 1 5
5 elemen pertama terurut Ascending, dan 5 elemen terakhir acak:
1 2 3 4 5 10 9 8 7 6
Semua elemen terurut ascending:
1 2 3 4 5 6 7 8 9 10
Semua elemen terurut Descending:
10 9 8 7 6 5 4 3 2 1
PS G:\My Drive\COLLAGE\FENS\SEMESTER 2\Algoritma & Struktur Data\M10-Sorting 1 (insertion &selection)\tugas\praktikum>

```

## 2. Insertion Sort

array 1: 3, 10, 4, 6, 8, 9, 7, 2, 1, 5	→ data awal, semua elemen terdistribusi acak
array 2: 1, 2, 3, 4, 5, 9, 7, 6, 8, 10	→ data awal, 5 elemen pertama terurut descending, dan 5 elemen terakhir acak
array 3: 1, 2, 3, 4, 5, 6, 7, 8, 9, 10	→ data awal, semua elemen terurut ascending
array 4: 10, 9, 8, 7, 6, 5, 4, 3, 2, 1	→ data awal, semua elemen terurut descending

Source Code :

```
#include <stdio.h>

int main()
{
    int arr[] = {3, 10, 4, 6, 8, 9, 7, 2, 1, 5};
    int temp = 0;

    // Hitung panjang array arr
    int length = sizeof(arr) / sizeof(arr[0]);

    // Menampilkan elemen array asli
    printf("Semua elemen terdistribusi acak : \n");
    for (int i = 0; i < length; i++)
    {
        printf("%d ", arr[i]);
    }

    // ascending order. 5 elemen terakhir acak
    for (int i = 0; i < length - 5; i++)
    {
        for (int j = i + 1; j < length; j++)
        {
            temp = arr[j];
            int k = j - 1;
            /* */
            while (k >= 0 && temp <= arr[k])
            {
                arr[k + 1] = arr[k];
                k = k - 1;
            }
            arr[k + 1] = temp;
        }
    }
    printf("\n");
    printf("5 elemen pertama terurut Ascending, dan 5 elemen terakhir acak: \n");
    for (int i = 0; i < length; i++)
    {
        printf("%d ", arr[i]);
    }

    // ascending order setelah 5 random
    for (int i = 1; i < length; i++)
    {
        temp = arr[i];
        int j = i - 1;

        while (j >= 0 && temp <= arr[j]) /* Move the elements greater than temp to
one position ahead from their current position*/
        {
            arr[j + 1] = arr[j];
            j = j - 1;
        }
    }
}
```

```

    }
    arr[j + 1] = temp;
}

printf("\n");
printf("data awal, semua elemen terurut ascending: \n");
for (int i = 0; i < length; i++)
{
    printf("%d ", arr[i]);
}

// descending order order
for (int i = 1; i < length; i++)
{
    temp = arr[i];
    int j = i - 1;
    /* */
    while (j >= 0 && temp >= arr[j])
    {
        arr[j + 1] = arr[j];
        j = j - 1;
    }
    arr[j + 1] = temp;
}

printf("\n");
printf("5 elemen pertama terurut Ascending, dan 5 elemen terakhir acak: \n");
for (int i = 0; i < length; i++)
{
    printf("%d ", arr[i]);
}
return 0;
}

```

Output :

```

insertion-sort.c - Algoritma & Struktur Data - Visual Studio Code
PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS G:\My Drive\COLLAGE\PENS\SEMESTER 2\Algoritma & Struktur Data> cd "g:\My Drive\COLLAGE\PENS\SEMESTER 2\Algoritma & Struktur Data\M10-Sorting 1 (insertion &selection)\tugas\praktikum\" ; if ($?) { gcc insertion-sort.c -o insertion-sort } ; if ($?) { . \insertion-sort }

INSERTION SORT
-----

Semua elemen terdistribusi acak :
3 10 4 6 8 9 7 2 1 5
5 elemen pertama terurut Ascending, dan 5 elemen terakhir acak:
1 2 3 4 5 6 7 8 9 10
data awal, semua elemen terurut ascending:
1 2 3 4 5 6 7 8 9 10
data awal, semua elemen terurut descending:
10 9 8 7 6 5 4 3 2 1
PS G:\My Drive\COLLAGE\PENS\SEMESTER 2\Algoritma & Struktur Data\M10-Sorting 1 (insertion &selection)\tugas\praktikum> 

```