

SORTING



ST3 Telkom Purwokerto

* Untuk kalangan sendiri

Praktikum 10
Materi : Sorting
Waktu : 100 menit

Tujuan

Setelah mengikuti praktikum ini, maka mahasiswa dapat:

1. Mahasiswa dapat melakukan perancangan aplikasi menggunakan struktur Sorting (Pengurutan)
2. Mahasiswa mampu melakukan analisis pada algoritma Sorting yang dibuat
3. Mahasiswa mampu mengimplementasikan algoritma Sorting pada sebuah aplikasi secara tepat dan efisien
4. Mahasiswa mampu menjelaskan mengenai algoritma Sorting
5. Mahasiswa mampu membuat dan mendeklarasikan struktur algoritma Searching
6. Mahasiswa mampu menerapkan dan mengimplementasikan algoritma Searching

Dasar Teori

Pengurutan data dalam struktur data sangat penting terutama untuk data yang bertipe data numerik ataupun karakter. Pengurutan dapat dilakukan secara ascending (urut naik) dan descending (urut turun). Pengurutan (Sorting) adalah proses pengurutan data yang sebelumnya disusun secara acak sehingga tersusun secara teratur menurut aturan tertentu.

Contoh:

Data Acak 3 9 6 21 10 1 13

Ascending 1 3 6 9 10 13 21

Descending 21 13 10 9 6 3 1

3 Metode Dasar pada Algoritma Sorting adalah sebagai berikut:

a. Insertion Sort

Mirip dengan cara orang mengurutkan kartu, selembat demi selembat kartu diambil dan disisipkan (insert) ke tempat yang seharusnya. Pengurutan **dimulai dari data ke-2** sampai dengan data terakhir, jika ditemukan data yang lebih kecil, maka akan ditempatkan (di-insert) diposisi yang seharusnya. Pada penyisipan elemen, maka elemen-elemen lain akan bergeser kebelakang.

Urutkan angka berikut menjadi angka terurut secara ASC

21	15	18	2	7	1
----	----	----	---	---	---

Proses 1

Temp	Cek	Geser
15	Temp < 21	Data ke-0 ke posisi 1

Temp menempati posisi ke-0

15	21	18	2	7	1
----	----	----	---	---	---

Proses 2

Temp	Cek	Geser
18	Temp < 21	Data ke-1 ke posisi 2
18	Temp > 15	

Temp menempati posisi ke-1

15	18	21	2	7	1
----	----	----	---	---	---

Proses 3

Temp	Cek	Geser
2	Temp < 21	Data ke-2 ke posisi 3
2	Temp < 18	Data ke-1 ke posisi 2
2	Temp < 15	Data ke-0 ke posisi 1

Temp menempati posisi ke-0

2	15	18	21	7	1
---	----	----	----	---	---

Proses 4

Temp	Cek	Geser
7	Temp < 21	Data ke-3 ke posisi 4
7	Temp < 18	Data ke-2 ke posisi 3
7	Temp < 15	Data ke-1 ke posisi 2
7	Temp > 2	-

Temp menempati posisi ke-1

2	7	15	18	21	1
---	---	----	----	----	---

Proses 5

Temp	Cek	Geser
1	Temp < 21	Data ke-4 ke posisi 5
1	Temp < 18	Data ke-3 ke posisi 4
1	Temp < 15	Data ke-2 ke posisi 3
1	Temp < 7	Data ke-1 ke posisi 2
1	Temp < 2	Data ke-0 ke posisi 1

Temp menempati posisi ke-1

1	2	7	15	18	21
---	---	---	----	----	----

b. Selection Sort

Selection sort merupakan kombinasi antara sorting dan searching. Setiap proses yang dilakukan akan dicari elemen – elemen yang belum diurutkan yang memiliki nilai terkecil atau terbesar akan ditukarkan ke posisi yang tepat dalam array. Contohnya, pada proses pengurutan pertama, akan dicari nilai terkecil dan data ini akan ditempatkan di indeks terkecil (data[0]), pada proses pengurutan kedua akan dicari data kedua terkecil, dan akan ditempatkan di indeks kedua (data[1]). Selama proses, perbandingan dan pengubahan hanya dilakukan pada indeks perbandingan saja, pertukaran data secara fisik terjadi pada akhir proses.

Urutkan angka berikut menjadi angka terurut secara ASC

0	1	2	3	4	5
21	15	18	2	7	1

Proses 1

Perbandingan	Status	Posisi indeks
21>15	Tukar indeks	1
15<18	-	-
15>2	Tukar indeks	3
2<7	-	-
2>1	Tukar indeks	5

Perubahan urutan data dengan menukar indeks 0 dan indeks 5

0	1	2	3	4	5
1	15	18	2	7	21

Proses 2

Perbandingan	Status	Posisi indeks
1<15	-	-
15<18	-	-
15>2	Tukar indeks	3
2<7	-	-
2<21	-	-

Perubahan urutan data dengan menukar indeks 1 dan indeks 3

0	1	2	3	4	5
1	2	18	15	7	21

Proses 3

Perbandingan	Status	Posisi indeks
1<2	-	-
2<18	-	-
18>15	Tukar indeks	3
15>7	Tukar indeks	4
7<21	-	-

Perubahan urutan data dengan menukar indeks 2 dan indeks 4

0	1	2	3	4	5
1	2	7	15	18	21

Proses 4

Perbandingan	Status	Posisi indeks
1<2	-	-
2<7	-	-
7<15	-	-
15<18	-	-
18<21	-	-

Tidak ada perubahan urutan data

0	1	2	3	4	5
1	2	7	15	18	21

c. Bubble Sort

Metode ini merupakan metode paling sederhana dan paling tidak efisien, karena memerlukan waktu yang relatif lebih lama dibandingkan dengan metode- metode yang lainnya. Konsep dasar dari Bubble sort ialah membandingkan elemen yang sekarang dengan elemen yang berikutnya, jika elemen sekarang > elemen berikutnya (untuk ascending), maka dilakukan proses penukaran. Proses sorting dapat dimulai dari data awal atau data akhir.

Urutkan angka berikut menjadi angka terurut secara ASC

0	1	2	3	4	5
21	15	18	2	7	1

Proses 1

0	1	2	3	4	5
21	15	18	2	7	1
15	21	18	2	7	1
15	18	21	2	7	1
15	18	2	21	7	1
15	18	2	7	21	1
15	18	2	7	1	21

Proses 2

0	1	2	3	4	5
15	18	2	7	1	21
15	2	18	7	1	21
15	2	7	18	1	21
15	2	7	1	18	21

Proses 3

0	1	2	3	4	5
15	2	7	1	18	21
15	2	1	7	18	21

Proses 4

0	1	2	3	4	5
15	2	1	7	18	21
2	15	1	7	18	21
2	1	15	7	18	21
2	1	7	15	18	21

Proses 5

0	1	2	3	4	5
2	1	7	15	18	21
1	2	7	15	18	21

GUIDED

1. Buatlah sebuah project dengan nama GD1_Kelas_NIM Program ini merupakan program untuk proses sorting dengan metode Insertion Sort

```
#include<iostream.h>
#include<conio.h>

main()
{
    int x,a[100],i,t,j,k;

    cout<<"Insertion Sort\n";
    cout<<"Masukkan banyak bilangan = ";cin>>x;
    for(i=1;i<=x;i++)
    {
        cout<<"\n Bilangan ke-"<<i<<" : ";cin>>a[i];
    }
    cout<<"\n\n data Sebelum diurutkan : ";
    for(i=1;i<=x;i++)
    {
        cout<<"    "<<a[i];
    }
    cout<<endl;
    for(i=1;i<=x;i++)
    {
        for(j=1;j<=i;j++)
        {

            if(a[i]>a[j])
            {
                t=a[i];
                a[i]=a[j];
                a[j]=t;
            }
        }
        cout<<"\n # "<<i<<" : ";
        for(k=1;k<=i;k++)
        {
            cout<<"    "<<a[k];
        }
    }
    cout<<"\n\n data Setelah diurutkan secara descending : ";
    for(i=1;i<=x;i++)
    {
        cout<<"    "<<a[i];
    }
    getch();
}
```


2. Buatlah sebuah project dengan nama GD2_Kelas_NIM yang merupakan sebuah program menu.

```
#include <stdio.h>
#include <conio.h>
#include <stdlib.h>

int data[10], data2[10];
int n;
void tukar (int a, int b) {
    int t;
    t = data[b];
    data[b] = data[a];
    data[a] = t;
}

void bubble_sort () {
    for (int i=1; i<n; i++) {
        for (int j=n-1; j>=i; j--) {
            if (data[j] < data[j-1])
                tukar(j, j-1);
        }
    }printf("bubble sort selesai!\n");
}

void selection_sort () {
    int pos, i, j;
    for (i=0; i<n-1; i++) {
        pos = i;
        for (j = i+1; j<n; j++) {
            if (data [j] < data[pos])
                pos = j;
        }
        if (pos != i) tukar (pos, i);
    }
    printf ("selection sort selesai!\n");
}
```

```

void insertion_sort () {
    int temp, i, j;
    for (i=1; i<n; i++) {
        temp = data[i];
        j = i-1;
        while (data[j] > temp && j>=0) {
            data[j+1] = data[j];
            j--;
        }
        data[j+1] = temp;
    }
    printf("insertion sort selesai!\n");
}

void Input () {
    printf("Masukkan jumlah data= ");
    scanf("%d",&n);
    for(int i=0;i<n;i++) {
        printf("Masukkan data ke-%d= ", (i+1));
        scanf("%d",&data[i]);
        data2[i]=data[i];
    }
}

void AcakLagi(){
    for(int i=0;i<n;i++){
        data[i]=data2[i];
    }
    printf("Data sudah teracak\n");
}

void Tampil(){
    printf("Data: ");
    for(int i=0;i<n;i++){
        printf("%d",data[i]);
    }
    printf("\n");
}

```

```

void main(){
//clrscr();
int pil;
do{
    //clrscr();
    printf ("1. Input Data\n");
    printf("2. Bubble Sort\n");
    printf("3. Selection Sort\n");
    printf("4. Tampilkan Data\n");
    printf("5. Acak\n");
    printf("0. Exit\n");
    printf("Pilihan = ");
    scanf("%d", &pil);
    switch(pil) {
        case 1: Input(); break;
        case 2: bubble_sort(); break;
        case 3: selection_sort(); break;
        case 4: Tampil(); break;
        case 5: AcakLagi(); break;
    }
    getch();
}
while(pil!=0);
}

```

Unguided

1. Buatlah program untuk mengurutkan karakter karakter yang dimasukkan sesuai dengan urutan alphabet

Misalnya:

Input:

f	z	h	t	u	q
---	---	---	---	---	---

Output:

f	h	q	t	u	z
---	---	---	---	---	---

Resume

PRAKTIKUM ALGORITMA DAN STRUKTUR DATA

S1 TEKNIK INFORMATIKA

Hari/Tanggal Praktikum :

Modul :

NIM :

Nama Praktikan :

Nama Asistant : 1.....

2.....

Nilai dan Parat :

Hasil Analisa Praktikum

[illegible]