

ALGORITMA PEMOGRAMAN

SEMESTER GENAP 2017/2018



ITERA

INSTITUT TEKNOLOGI SUMATERA

2018

MODUL 4a

Petunjuk Praktikum

Modul ini dilaksanakan dalam 1 (satu) sesi praktikum.

Tiap sesi praktikum dilaksanakan dalam 3 tahap yang dilaksanakan secara berturut-turut, yaitu:

1. Materi

Tujuan	Me-review materi-materi yang telah diajarkan di kelas untuk memudahkan mahasiswa memahami soal-soal yang akan diberikan.
Durasi	±25 menit
Pelaksanaan	Asisten akan menjelaskan topik pembahasan modul ini, serta memberikan contoh-contoh terkait yang dapat memudahkan mahasiswa memahami konsep-konsep pemrograman. Mahasiswa dapat bertanya kepada asisten mengenai topik pembahasan yang dijelaskan asisten.

2. Tugas Praktikum

Tahap ini wajib dilakukan di lab!

Tujuan	Mahasiswa mengerjakan tugas-tugas praktikum secara mandiri sesuai dengan topik yang diberikan dan mengumpulkan hasilnya untuk penilaian
Durasi	±80 menit
Sifat	Individual (tidak diperkenankan bekerja sama dengan pihak mana pun), tetapi masih boleh membuka material kuliah miliknya sendiri
Pelaksanaan	Mahasiswa mengerjakan soal-soal praktikum untuk modul terkait yang terdapat pada bab TUGAS PRAKTIKUM sesuai dengan petunjuk yang ada dalam durasi yang ditetapkan.

3. Post Test

Tujuan	Mahasiswa mengulas kembali materi yang telah dilakukan selama praktikum dengan soal-soal yang diberikan asisten. Post test ini dapat digunakan sebagai tolak ukur capaian materi yang didapat oleh mahasiswa.
Durasi	±15 menit
Sifat	Individual (tidak diperkenankan bekerja sama dengan pihak mana pun), dan tidak boleh membuka material kuliah miliknya sendiri
Pelaksanaan	Mahasiswa mengerjakan soal-soal sebagai review dari pembahasan dan latihan modul ini.

Petunjuk Penamaan dan Penulisan File Program

1. Pada setiap soal tugas praktikum, perhatikan petunjuk penamaan file.
2. Pada petunjuk penamaan file, gantilah <NIM> dengan NIM Anda masing-masing dan XX dengan nomor soal dalam 2 digit.

Contoh:

Jika file harus disimpan dengan format: **P1_<NIM>_XX.cpp** dan NIM Anda adalah **123456**

serta nomor soal yang sedang dikerjakan adalah **2**, maka nama file Anda adalah:

P1_123456_02.cpp

3. Untuk setiap file source code program berikan identitas, minimum:

```
// NIM>Nama :
```

```
// Nama file :
```

```
// Tanggal :
```

```
// Deskripsi :
```

4. Simpan dan upload file source code hasil latihan dan praktikum pada direktori yang ditentukan asisten.

Pengertian Sorting

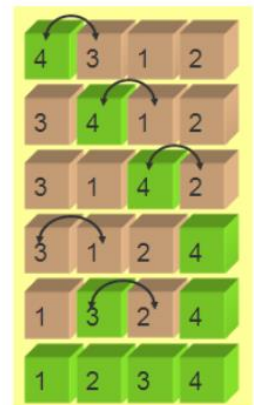
Sorting yaitu pengurutan data-data baik dari data yang terkecil ke terbesar, maupun sebaliknya. Proses pengurutan banyak ditemukan dalam komputer. Hal ini karena data yang sudah urut akan lebih cepat untuk dicari. Untuk membentuk data yang tidak urut menjadi data yang urut, terdapat berbagai metode dan algoritma yang bisa digunakan.

Metode Bubble Sort

Merupakan algoritma pengurutan yang paling tua. Cara pengurutan ini adalah dengan membandingkan masing-masing item dalam suatu array dan menukarkan item jika dibutuhkan (ketika item lebih besar / lebih kecil dari array yang dibandingkan).

Cara pengurutan dengan metode bubble sort

- Mulai dari elemen ke-1:
 1. Jika $s_2 < s_1$, pertukarkan
 2. Jika $s_3 < s_2$, pertukarkan
 - ...
 3. Jika $s_{n-1} < s_n$, pertukarkan
- Ulangi lagi untuk *pass* ke-2, 3, ..., $n - 1$ dst
- Semuanya ada $n - 1$ kali *pass*



Sumber gambar: Prof. Amr Goneid
Department of Computer Science, AUC

Pseudo code bubble sort

```
procedure BubbleSort (input/output s : TabelInt, input n : integer)
{ Mengurutkan tabel s[1..N] sehingga terurut menaik dengan metode
  pengurutan bubble sort.

  Masukan : Tabel s yang sudah terdefinisi nilai-nilainya.
  Keluaran: Tabel s yang terurut menaik sedemikian sehingga
    s[1] ≤ s[2] ≤ ... ≤ s[N].
}
Deklarasi
  i      : integer      { pencacah untuk jumlah langkah }
  k      : integer      { pencacah, untuk pengapungan pada setiap
langkah }
  temp   : integer      { peubah bantu untuk pertukaran }

Algoritma:
  for i ← n-1 downto 1 do
    for k ← 1 to i do
      if s[k+1] < s[k] then
        {pertukarkan s[k] dengan s[k+1]}
        temp ← s[k]
        s[k] ← s[k+1]
        s[k+1] ← temp
      endif
    endfor
  endfor
```

Metode Insertion Sort

Pengurutan dengan Metode Insertion Sort yaitu dengan penyisipan adalah suatu metode yang melakukan pengurutan dengan cara menyisipkan data yang belum urut ke dalam bagian data yang telah terurut secara relatif, penyisipan dilakukan ke bagian sisi kiri. penyisipan yang dilakukan ini menimbulkan pergeseran data ke bagian kanan array.

Ilustrasi :

Data belum terurut :

data				
0	1	2	3	4
8	10	7	12	11

Data[1] dibandingkan dengan data[0]. $\text{Data}[1] > \text{data}[0]$ maka tidak perlu ada pengurutan.

data				
0	1	2	3	4
8	10	7	12	11

Data[2] dibandingkan dengan data[1]. $\text{Data}[2] < \text{data}[1]$, maka data[2] diurutkan ke dalam array yang telah terurut (array sebelah kiri/berwarna abu-abu)

data				
0	1	2	3	4
8	10		12	11

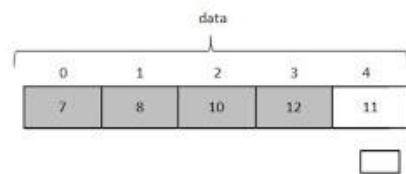
7

data				
0	1	2	3	4
	8	10	12	11

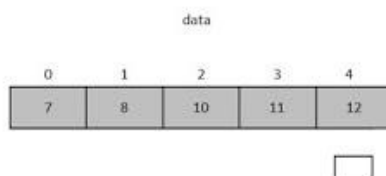
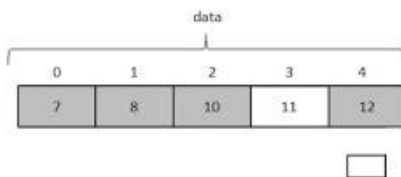
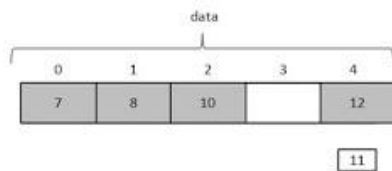
7

data				
0	1	2	3	4
7	8	10	12	11

Data[3] dibandingkan dengan data[2]. Data[3] > data[2] maka tidak perlu ada pengurutan.



Data[4] dibandingkan dengan data[3]. Data[4] < data[3], maka data[4] diurutkan ke dalam array yang telah terurut.



Pseudo code insertion sort

```
Insertion Sort (A, n)
{
    for i <- 1 to n-1
    {
        value <- A[i]
        index <- i
        while (index > 0 && A[index-1] > value)
        {
            A[index] <- A[index-1]
            index <- index-1
        }
        A[index] <- value
    }
}
```