

Algoritma dan Pemrograman Lanjut

Pertemuan Ke-1 Pengantar Algoritma dan Pemrograman Lanjut



Disusun Oleh :
Wilis Kaswidjanti, S.Si., M.Kom.

**Jurusan Teknik Informatika
Fakultas Teknologi Industri
Universitas Pembangunan Nasional “Veteran”
Yogyakarta**

Algoritma dan Pemrograman Lanjut

Judul Materi : Pengantar Algoritma dan Pemrograman Lanjut
(Review Algo)

Deskripsi Materi :

Tujuan Instruksional Khusus :

Referensi :

- Buku Teks
Munir, Rinaldi (2005), *Algoritma dan Pemrograman dalam Bahasa Pascal dan C*, Buku 1, Edisi Ketiga, Penerbit Informatika Bandung.
Charibaldi, N. (2004), *Modul Kuliah Algoritma Pemrograman I*, Edisi Kedua, Yogyakarta
- Buku Acuan/Referensi
Brassard, Gilles (1999), *Fundamentals of algorithma*, PrinteceHall.
Jarne, Stroustrup B. (1997), *C++ Programming language*, AT &T.
Kristanto, Andri (2003), *Algoritma pemrograman C++*, Graha Ilmu.
Schildt,Herbert (2000), *The Complete Reference C++*, McGraw-Hill.
Sedgewick, R. (2000), *Algoritma Third edition In C part 5*, Addison Wesley.

PENGANTAR ALGORITMA DAN PEMROGRAMAN LANJUT

PENDAHULUAN

Materi perkuliahan Algoritma dan Pemrograman 2 merupakan kuliah lanjutan dari perkuliahan Algoritma dan Pemrograman 1. Berikut pengantar untuk mereview kembali materi yang harus sudah dikuasai untuk mengikuti perkuliahan Algoritma dan Pemrograman 2.

ISI

1. Algoritma dan Program

- Algoritma adalah urutan logika langkah kerja untuk menyelesaikan suatu masalah.
- Program adalah logika pemecahan masalah dalam bahasa pemrograman tertentu untuk diproses oleh computer.
- Belajar memrogram berarti belajar strategi pemecahan masalah metodologi dan sistematika pemecahan masalah.
- Belajar bahasa pemrograman berarti belajar memakai suatu bahasa pemrograman tertentu.
- Dasar-dasar setiap algoritma walaupun sekomplek apapun meliputi runtunan, pemilihan dan pengulangan.
- Cara pendekatan *Top Down Design* sangat bermanfaat dalam membuat algoritma untuk masalah yang cukup rumit dan kompleks

2. Notasi Algoritma

Pada dasarnya algoritma selalu disusun berdasarkan 3 bagian, antara lain Judul Algoritma, Deklarasi/kamus dan Deskripsi/algoritma. Agar dapat dilaksanakan oleh computer, maka algoritma harus ditranslasikan ke dalam suatu notasi bahasa pemrograman. Agar notasi algoritma mudah ditranslasikan kedalam notasi bahasa pemrograman, maka sebaliknya notasi algoritmik tersebut berupa *pseudo-code* yang berkoresponden dengan bahasa pemrograman secara umum.

3. Operator, identifier dan tipe data

- Operator adalah notasi yang dipakai untuk melaksanakan suatu operasi terhadap data dan identifier.
- Identifier adalah sesuatu yang dipakai sebagai nama pengenalan.
- Tipe data pada sebuah objek merupakan himpunan nilai yang dapat dimilikinya dan operasi yang dapat dilakukan pada objek tersebut.

3. Input dan Output

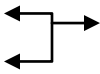
Notasi input dan output sangat penting dalam suatu pemrograman karena merupakan sarana interaksi antara komputer dan manusia dalam berkomunikasi. Banyak format input dan output yang dimiliki bahasa pemrograman C++, tinggal programmer yang menentukan format mana yang dipakai sesuai kebutuhan.

4. Statemen Pengendalian

Pada dasarnya ada dua macam statement pengendalian yaitu IF-THEN-ELSE dan CASE-OF. Pada prakteknya dua macam notasi tersebut dapat dimodifikasi sesuai kebutuhan.

5. Statemen Pengulangan

Tiga macam notasi struktur pengulangan :

1. Struktur FOR atau TRAVERSAL (*Pengulangan tanpa kondisi / Unconditional Looping*).
 2. Struktur WHILE..DO
 3. Struktur REPEAT..UNTIL
- 
- (*Pengulangan dengan kondisi / Conditional Looping*).

Tiga macam notasi struktur pengulangan dalam bahasa C++ :

1. FOR
2. WHILE
3. DO..WHILE

6. Prosedur dan Fungsi

Procedure dan function di dalam algoritma memiliki perbedaan. Tetapi di bahasa pemrograman C++ semua disebut function dengan perbedaan sintak untuk membedakan antara fungsi sebagai procedure dan sebagai function yang dimaksud pada algoritma

Struktur Prosedur :

1. Bagian Judul (header).

Terdiri atas **Nama prosedur dan deklarasi parameter (jika ada)**.

2. Bagian deklarasi.

Untuk mengumumkan nama-nama.

3. Bagian deskripsi.

Disebut juga **badan prosedur**.

Adapun bentuk umum definisi sebuah fungsi (dalam bahasa C++) adalah :

```
penentu-tipe nama_fungsi(daftar parameter)
{
    deklarasi variable lokal
    tubuh fungsi
}
```

Penentu tipe adalah untuk menentukan tipe keluaran fungsi yang dapat berupa satu tipe data C++ yang berlaku, misalnya char atau int. Default tipe fungsi yang tidak disebutkan dianggap sebagai int.

7. Mesin Abstrak

Mesin abstrak adalah mesin yang dianggap ada dan diasumsikan mampu melakukan mekanisme yang didefinisikan untuk mesin tersebut. Mesin gambar adalah abstrak yang terdiri dari 2 unit yaitu pena dan bidang gambar.

8. Tipe Data Array

Tipe data array adalah salah satu tipe terstruktur dalam pemrograman. Tipe data array ini dapat dipakai untuk salah satu contohnya adalah perhitungan matriks.

PENUTUP

Materi Algoritma dan Pemrograman 2 memakai semua yang telah diajarkan pada matakuliah Algoritma dan Pemrograman 1, seperti algoritma, tipe data, operator, statement pengendalian, statement pengulangan yang merupakan dasar dari pemrograman.

SOAL-SOAL

1. Apa **output/keluaran** algoritma atau program pada soal (1), (2), (3) dan (4) berikut (pilih a, b, c, d atau e) :

```
(1) Algoritma Salam
KAMUS
  A,B,C : integer
  S      : string
ALGORITMA
A ← 100;
B ← 100;
IF (A < B) THEN
  C ← B - A;
  IF (C = A) THEN S ← "Hallo"
  ELSE S ← "Apa"
  ENDIF
ELSE IF (B > A) THEN S ← "Khabar"
  ELSE S ← "Disana"
  ENDIF
ENDIF
Output(S)
```

a.Hallo **b.**Apa **c.**Khabar **d.**Disana

```
(2) #include <stdio.h>
int Pilihan;

main()
{
  Pilihan = 8 % 3;
  switch (Pilihan) {
    case 1 :
      printf("%c", 'A');
    case 2 :
      printf("%c", 'B');
    case 3 :
      printf("%c", 'C');
    default:
      printf("%c", '$');
  }
}
```

a.\$ **b.**B **c.**C **d.**ABC\$ **e.**BC\$

```
(3) Algoritma Looping
KAMUS
  kondisi : boolean
  i        : integer
ALGORITMA
i ← 100
REPEAT
  i ← i - 30
  kondisi ← (i < 0)
UNTIL (kondisi)
Output(i)
```

a. 0 **b.** 70 **c.** -20 **d.** -10

```
(4) #include <stdio.h>
int A, i;

main()
{
  A = 0;
  for (i = 1; i <= 10; i++)
  {
    A = A + i + 3;
    if (A > 15) break;
  }
  printf("%d", i);
}
```

a. 1 **b.** 2 **c.** 3 **d.** 4 **e.** 5

2. Apakah output/keluaran dari program berikut :

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

main()
{
    int i;
    int x[9]={1,2,3,0,7,0,1,8,5};

    for(i=8;i>=0;i--)
    {
        if (x[i]<5)
            cout<<"x["<<i<<"]="<<x[i]<<endl;
    }
}
```

2. Ubahlah *penggalan* algoritma di bawah ini ke dalam bahasa C++

(dengan menggunakan sintaks pengulangan **do-while**)

```
for p←1 to 4 do
    for q←1 to p do
        output(q)
    endfor
endfor
```