

PENDAHULUAN

TUJUAN

Tujuan dari belajar Algoritma dan Pemrograman 1A atau Flowchart adalah agar dapat membiasakan diri melakukan suatu perencanaan apabila menyelesaikan suatu masalah, karena suatu permasalahan yang diselesaikan dengan suatu perencanaan yang matang maka akan mendapatkan solusi yang lebih optimal dibandingkan menyelesaikan masalah tanpa menggunakan suatu perencanaan.

KEGUNAAN FLOWCHART

Fungsi dari Flowchart adalah untuk membantu dalam pembuatan program secara umum setelah itu dituangkan ke dalam program secara detail. Sehingga memudahkan pembuat program dan menghasilkan program yang terstruktur serta output yang sesuai dengan perencanaan

KONSEP PEMROGRAMAN

Program adalah kata, ekspresi,, pernyataan atau kombinasi yang disusun dan dirangkai menjadi satu kesatuan prosedur yang berupa urutan langkah untuk menyelesaikan masalah dan diimplementasikan dengan menggunakan bahasa pemrograman sehingga dapat dieksekusi oleh komputer

Pemrograman adalah proses mengimplementasikan urutan langkah untuk menyelesaikan suatu masalah dengan menggunakan suatu bahasa pemrograman

STANDAR PROGRAM YANG BAIK

Standar pemrograman dibutuhkan untuk menciptakan suatu program yang baik yang memiliki portabilitas yang tinggi, sehingga memudahkan dalam merancang dan merawat program serta meningkatkan efektifitas penggunaan peralatan komputer.

Untuk menentukan standar program yang baik dibutuhkan beberapa standar sebagai dasar penilaian, seperti :

- a. Pemecahan masalah**
- b. Penyusunan program**
- c. Perawatan program**
- d. Standar prosedur**

Standar-standar tersebut sering dilihat oleh pemrogram sebagai batasan kreatifitas dan kemampuan untuk menuangkan berbagai ide ke dalam bentuk program. Namun dengan adanya standar akan membuat program menjadi konsisten dan mudah untuk dikembangkan.

STANDAR TEKNIK PEMECAHAN MASALAH

Setelah masalahnya dipahami dengan baik, seorang pemrogram tentu membutuhkan suatu teknik untuk memecahkan masalah tersebut, antara lain dikenal **teknik Top-Down** dan **teknik Bottom-Up**

Teknik Top-Down

Merupakan teknik pemecahan masalah yang paling umum digunakan. Pada teknik ini, suatu masalah yang kompleks dibagi-bagi ke dalam beberapa kelompok masalah yang lebih kecil. Dari kelompok masalah yang kecil tersebut di analisis. Apabila dimungkinkan maka masalah tersebut akan dipilah lagi menjadi sub bagian dan setelah itu mulai disusun langkah-langkah untuk menyelesaikan secara detail.

Teknik Bottom Up

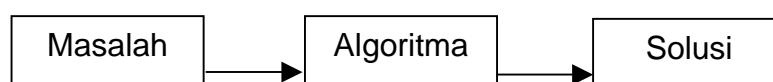
Merupakan teknik pemecahan masalah yang mulai ditinggalkan, karena sulit untuk melakukan standarisasi proses dari prosedur-prosedur yang sudah terbentuk yang akan digabungkan. Pada teknik ini, bila ada masalah kompleks, maka pemecahan masalah dilakukan dengan menggabungkan prosedur-prosedur yang ada menjadi satu kesatuan program guna menyelesaikan masalah tersebut.

Setelah memilih teknik pemecahan masalah yang akan digunakan, pemrogram akan mulai menyusun langkah-langkah untuk memecahkan masalah secara detail yang disebut algoritma

Algoritma berasal dari kata **algoris** dan **ritmis** yang pertama kali diungkapkan oleh Abu Ja'far Mohammed Ibn Musa al Khowarizmi (825 M) dalam buku Al-Jabr Wa-al Muqabla.

Dalam bidang pemrograman, **Algoritma** didefinisikan sebagai suatu metode khusus yang tepat dan terdiri dari serangkaian langkah yang terstruktur dan dituliskan secara sistematis yang akan dikerjakan untuk menyelesaikan suatu masalah dengan bantuan komputer.

Hubungan antara algoritma, masalah dan solusi dapat digambarkan sebagai berikut :



Proses dari masalah hingga terbentuk suatu algoritma disebut tahap pemecahan masalah, sedangkan tahap dari algoritma hingga terbentuk suatu solusi disebut dengan tahap implementasi. Solusi yang dimaksud adalah suatu program yang merupakan implelementasi dari algoritma yang disusun.

Algoritma pemrograman yang baik memiliki ciri-ciri sebagai berikut :

- memiliki logika perhitungan/metode yang tepat dalam memecahkan masalah
- menghasilkan output yang tepat dan benar dalam waktu yang singkat
- ditulis dengan bahasa yang standar secara sistematis dan rapi sehingga tidak menimbulkan arti ganda
- ditulis dengan format yang mudah dipahami dan diimplementasikan ke dalam bahasa pemrograman
- semua operasi yang dibutuhkan terdefinisi dengan jelas
- semua proses harus selalu berakhir setelah sejumlah langkah dilakukan

CONTOH

Algoritma ketika ingin mengirimkan surat kepada teman, yaitu :

1. Tulis surat pada secarik kertas surat
2. Ambil sampul surat
3. Masukkan surat ke dalam sampul
4. Tutup sampul surat menggunakan perekat
5. Tempel perangko pada surat
6. Bawa surat ke kantor pos untuk di poskaan

STANDAR PENYUSUNAN PROGRAM

Dalam menyusun program, ada beberapa kriteria yang harus diperhatikan oleh seorang pemrogram, misalnya :

a. Kebenaran logika dan penulisan

Program yang disusun harus memiliki kebenaran logika pemecahan masalah maupun penulisan. Program harus memiliki ketepatan, ketelitian dan kebenaran dalam perhitungan sehingga hasilnya dapat dipercaya

Dalam penyusunan program, pemrogram tidak boleh hanya berpegang pada prinsip “asal program dapat dieksekusi saja”, tetapi harus benar-

benar teliti dalam menulis rumus-rumus dan urutan logis dari langkah-langkah pemecahan masalah yang disusun

b. Waktu minimum untuk penulisan program

Dalam penulisan program, pemrogram harus dapat menentukan waktu minimum penulisan programnya. Waktu minimum penulisan program adalah waktu yang harus tersedia secara wajar untuk menyusun program, dari awal hingga siap dioperasikan

c. Kecepatan maksimum eksekusi program

Ada beberapa faktor yang harus diperhatikan untuk dapat menghasilkan program yang memiliki kecepatan eksekusi maksimum, antara lain bahasa pemrograman yang digunakan (basis interpreter atau compiler), algoritma yang disusun, teknik pemrograman yang diterapkan dan perangkat keras yang dipakai untuk mengoperasikannya.

Kecepatan maksimum eksekusi program juga dapat ditingkatkan dengan memperbaiki struktur program, misalnya dalam proses pengujian. Hindilah proses pengujian yang berulang-ulang secara percuma.

Contoh:

If item = nilai1

Instruksi1

Endif

If item = nilai 2

Instruksi2

Endif

If item = nilai 3

Instruksi3

Endif

Bentuk di atas merupakan bentuk sequence dari tiga kali pengujian untuk mendapatkan satu alternatif. Lebih baik jika bentuk pengujian diatas ditulis sebagai berikut :

```
If item = nilai1  
    Instruksi1  
Else  
    If item = nilai 2  
        Instruksi2  
    Else  
        If item = nilai 3  
            Instruksi3  
        Endif  
    Endif  
Endif
```

Bentuk kedua di atas menyebabkan nilai item belum tentu di uji sebanyak tiga kali, dan segera setelah pengujian berhasil mengambil keputusan, maka proses pengujian tidak dilanjutkan.

d. Eksresi penggunaan memori

Seorang pemrogram perlu mempelajari teknik-teknik pembuatan program yang meminimumkan penggunaan memori. Pemborosan pemakaian memori akan menyebabkan eksekusi berjalan lambat.

e. Kemudahan merawat dan mengembangkan program

Program hendaknya memiliki struktur pemrograman yang baik, struktur data yang jelas dan dilengkapi dengan dokumentasi sehingga mudah untuk dipahami, diuji dan dikembangkan

f. User Friendly

Program yang disusun harus memiliki fasilitas-fasilitas yang memberikan kemudahan bagi pemakai untuk mengoperasikannya, misalnya dengan penambahan fasilitas on line help guna memberi penjelasan jika terjadi kesulitan, menu pilihan, tampilan yang informatif, pesan-pesan yang sederhana dan singkat sehingga mudah untuk dipahami, dan sebagainya

g. Portability

Usahakan agar program yang disusun dapat dioperasikan dengan berbagai jenis system operasi dan perangkat keras yang berbeda, sehingga fleksibel untuk digunakan

h. Pemrograman Modular

Pada **teknik top down**, masalah yang besar dan kompleks di bagi-bagi ke dalam beberapa kelompok masalah yang lebih kecil. Kelompok masalah kecil itu disebut **modul** dan teknik pemrograman terstruktur yang digunakan untuk mengimplementasikan langkah-langkah pemecahan masalah pada kelompok masalah yang kecil tersebut dikenal dengan sebutan teknik pemrograman modular. Namun setelah masing-masing modul disusun maka harus dibuat suatu system untuk mengintegrasikannya sehingga menjadi satu kesatuan program yang lengkap.

MEMBANGUN PROGRAM

Bila masalah yang dihadapi oleh seorang pemrogram sangat besar dan kompleks, maka ia pasti membutuhkan tahapan-tahapan dalam membangun programnya, agar dapat diperoleh suatu program yang baik.

Sebenarnya tahapan-tahapan tersebut hanya membantu pemrogram agar dapat menyelesaikan pekerjaannya secara sistematis.

Tahapan-tahapan untuk membangun suatu program yang besar dan kompleks adalah sebagai berikut :

- a. Definisi masalah**
- b. Analisis kebutuhan**
- c. Teknik pemecahan masalah dan algoritma**
- d. Pengkodean**
- e. Testing dan Debuging**
- f. Pemeliharaan**
- g. Dokumentasi**

Untuk memecahkan suatu masalah yang sederhana pemrogram tidak perlu melalui tujuh tahap tersebut, tetapi cukup mengidentifikasi masalah, menentukan input, proses dan output yang diinginkan, menentukan algoritma, mengimplementasikannya dengan sautu bahasa pemrograman tertentu dan melakukan testing.

ALASAN MENGGUNAKAN TAHAP-TAHAP PENYUSUNAN PROGRAM

Ada beberapa alasan mengapa seorang pemrogram perlu menggunakan tahap-tahap penyusunan program, yaitu :

a. Pertimbangan Logis

Melalui tahap definisi masalah dan analisis kebutuhan, maka pemrogram dapat mengetahui dengan jelas bobot atau tingkat kesulitan yang harus dipecahkan sehingga dapat melakukan perhitungan yang tepat untuk menentukan jumlah tenaga, harga, sumber daya dan waktu yang dibutuhkan untuk menyusun program

b. Pertimbangan analogi

Menyusun program seakan-akan identik dengan membangun rumah dimana dibutuhkan rancangan arsitektur dan definisi tahap-tahap pengerjaan yang terencana secara baik dan sistematis, sebelum ia mulai membangun

c. Pertimbangan jumlah data

Jumlah dan jenis data serta proses pengolahan data merupakan suatu hal yang perlu diperhatikan secara khusus. Jumlah dan jenis data sangat mempengaruhi dalam menentukan jenis variable dan yang akan digunakan dalam penyusunan program tersebut

d. Pertimbangan untuk pengembangan

Melalui tahap-tahap penyusunan program akan diperoleh suatu program yang baik, terstruktur dan sistematis, sehingga dapat dengan mudah dikembangkan di masa mendatang

e. Pertimbangan prinsip

Dalam menyusun program sebaiknya pemrogram membuat persiapan dan rancangan arsitektur program dengan matang, sehingga tidak menimbulkan masalah pada tahap coding dan testing