

Mata Kuliah : Dasar Dasar Pemrograman (Teori)  
Kode Mata Kuliah : KBTI4104  
Waktu : Selasa (10.40 – 12.40)  
Jumlah SKS : 4 SKS  
Nama Dosen : Ade Chandra  
Minggu ke : 7 (Tujuh)  
Tanggal : 27-10-2015  
Judul Materi : Struktur Proses

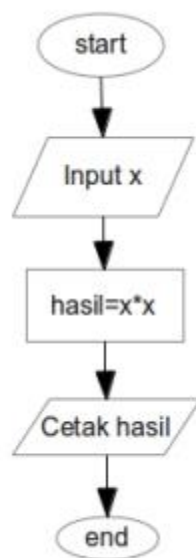
### Struktur PRIMITIF

Struktur primitive adalah struktur yang tidak dapat dinyatakan melalui struktur lainnya (tidak bisa diuraikan lagi). Struktur primitif merupakan komponen dasar pembentukan struktur algoritma. Struktur primitive dibagi menjadi 3 macam struktur primitive non parallel yaitu mulai dari proses *sequencial* proses pemilihan (*selection process*) dan proses perulangan (*repetition process*). Proses-proses ini termasuk kedalam *konstruksi atau struktur dasar algoritma*.

### Sequence process

Proses Sequence adalah sederetan pernyataan yang urutan pelaksanaannya / eksekusinya dilakukan secara beruntun dari baris pertama, kedua dan seterusnya hingga baris terakhir.

Jika diaplikasikan kedalam sebuah diagram alir (flowchart) maka akan menjadi seperti gambar di bawah ini.

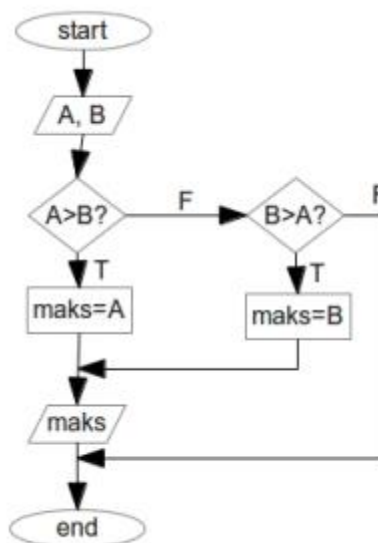


proses Sequence

flowchart di atas menggambarkan proses mencari nilai kuadrat dari bilangan x yang diinput. Proses tersebut dijalankan secara berurutan mulai dari input x, mengalikan x dan menyimpannya di variabel hasil selanjutnya variable hasil tersebut akan dicetak.

## Selection process ( proses pemilihan )

Saat algoritma kita dijalankan ada saat dimana suatu instruksi baru boleh dijalankan jika kondisi tertentu terpenuhi. Contohnya saat kita menerima raport atau nilai semester, disana akan ada kondisi yang menyatakan jika nilai kita lebih besar dari 80 maka kita mendapat A, jika nilai lebih besar dari 70 maka kita mendapat B, dan seterusnya. Kasus pemilihan seperti ini sudah biasa kita alami di kehidupan sehari-hari, namun bagaimana jika kita ingin *mengimplementasikannya* kedalam bentuk *algoritma*? . Dibawah ini merupakan contoh proses pemilihan untuk menentukan ukuran dua buah bilangan yang saya buat kedalam bentuk diagram alir.



proses seleksi

Penjelasan :

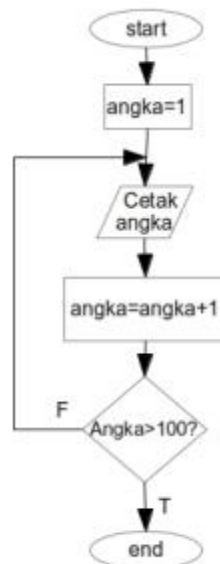
Pada awal proses kita akan menginput nilai untuk variable A dan variable B. setelah itu nilai yang telah kita input akan diseleksi jika nilai A lebih besar dari B maka nilai A akan diisikan kedalam variable lain yang bernama maks, sedangkan jika nilai A tidak lebih besar dari B maka akan diseleksi lagi dan jika nilai B yang lebih besar dari A maka nilai B akan diisikan kedalam variable maks. Variable maks yang telah berisi nilai A atau B tersebut nantinya akan ditampilkan. Namun jika nilai B juga tidak lebih besar dari A itu artinya nilai A dan B adalah sama dan komputer tidak akan menampilkan output apa-apa.

## Repetition process ( proses perulangan )

Salah satu manfaat penggunaan komputer adalah kemampuan mengulang sesuatu dengan cepat dan akurat. Kondisi perulangan seperti ini sering kita jumpai pada kehidupan sehari-hari seperti bangun tidur, sarapan setiap pagi, berangkat sekolah, pulang hingga tidur kembali.

Dalam algoritma, struktur perulangan ini sangat bermanfaat jika kita ingin melakukan sebuah proses yang sama berulang-ulang dengan kondisi tertentu, contohnya jika kita ingin program yang kita buat terus berjalan selama pengguna tidak memilih pilihan keluar, contoh sederhana lainnya adalah jika kita ingin mencetak bilangan dari 1 sampai 100 dan masih banyak lagi yang lainnya.

Untuk lebih mempermudah pemahaman anda, dibawah ini merupakan gambar flowchart yang menggunakan struktur perulangan untuk menampilkan bilangan 1 sampai 100.



proses looping

Penjelasan :

pada flowchart diatas kita memiliki variabel angka yang terlebih dahulu kita isi dengan nilai 1, setelah itu komputer akan mencetak variabel angka tersebut dan mengisi angka dengan nilai hasil penjumlahan angka tersebut dengan nilai 1 ( $\text{angka} = \text{angka} + 1$ ). Setelah itu kondisi akan diperiksa untuk mengetahui apakah angka memiliki nilai yang lebih besar dari 100 atau tidak, jika tidak komputer akan kembali untuk mencetak angka dan melakukan penjumlahan lagi sedangkan jika tidak maka proses akan berhenti.

Pada contoh kasus diatas menunjukan bahwa proses akan terus dilakukan selama kondisi belum terpenuhi.