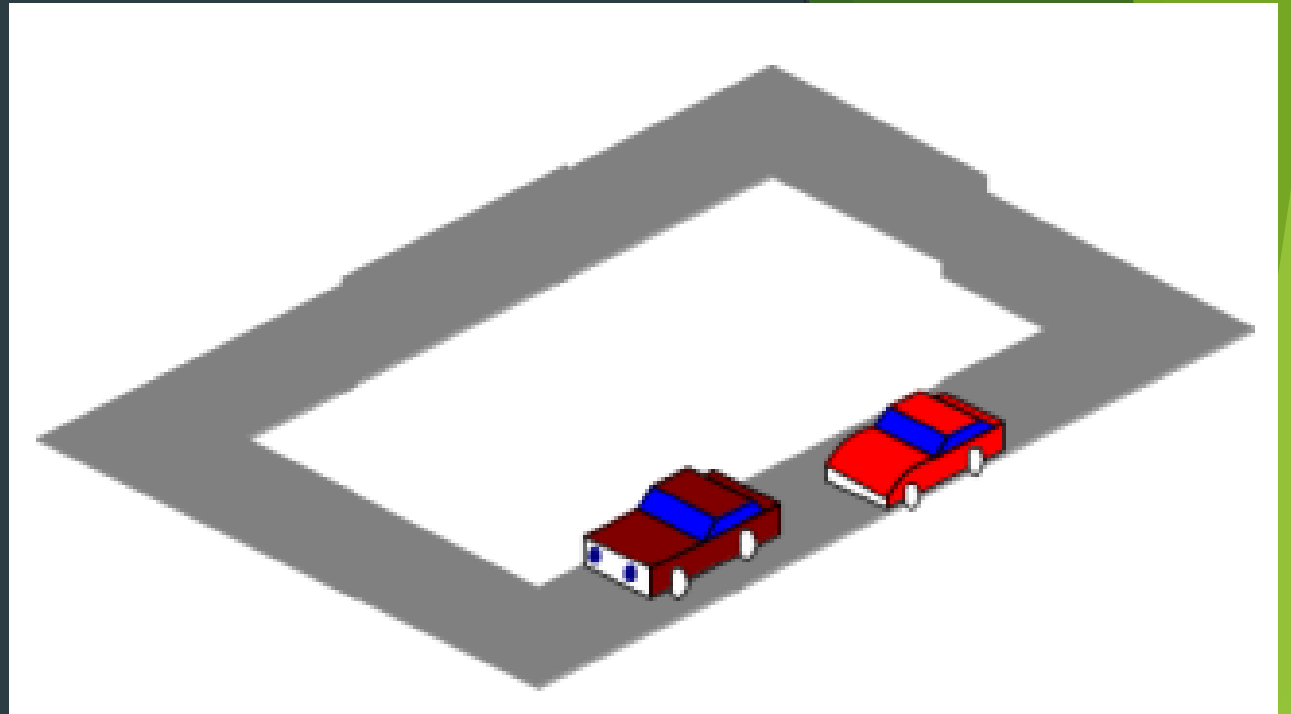


Struktur algoritma pengulangan

- Dalam banyak kasus seringkali kita dihadapkan pada sejumlah pekerjaan yang harus diulang berkali. Salah satu contoh yang gampang kita jumpai adalah balapan mobil seperti tampak pada gambar. Mobil-mobil peserta harus mengelilingi lintasan sirkuit berkali-kali sesuai yang ditetapkan dalam aturan lomba. Siapa yang mencapai garis akhir Paling cepat, dialah yang menang.



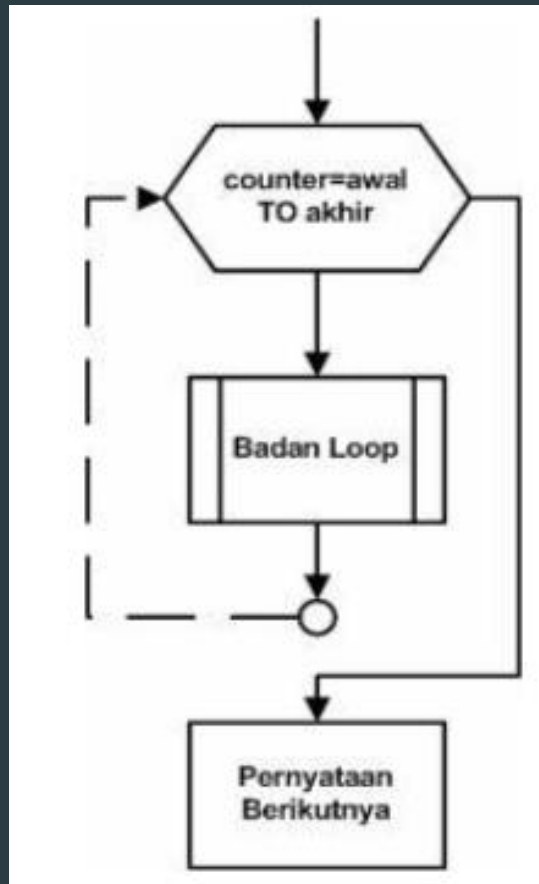
Gambar Lomba Balap Mobil di Sirkuit

- ▶ Pada pembuatan program komputer, kita juga kadang-kadang harus mengulang satu atau sekelompok perintah berkali-kali agar memperoleh hasil yang diinginkan. Dengan menggunakan komputer, eksekusi pengulangan mudah dilakukan. Hal ini karena salah satu kelebihan komputer dibandingkan dengan manusia adalah kemampuannya untuk mengerjakan tugas atau suatu instruksi berulang kali tanpa merasa lelah, bosan, atau malas. Bandingkan dengan pengendara mobil balap, suatu ketika pasti dia merasa lelah dan bosan untuk berputar-putar mengendarai mobil balapnya.
- ▶ Struktur pengulangan terdiri dari dua bagian :
 1. **Kondisi pengulangan**, yaitu syarat yang harus dipenuhi untuk melaksanakan pengulangan. Syarat ini biasanya dinyatakan dalam ekspresi Boolean yang harus diuji apakah bernilai benar (true) atau salah (false)
 2. **Badan pengulangan (loop body)**, yaitu satu atau lebih instruksi yang akan diulang

- ▶ Pada struktur pengulangan, biasanya juga disertai bagian inisialisasi dan bagian terminasi.
- ▶ **Inisialisasi** adalah instruksi yang dilakukan sebelum pengulangan dilakukan pertama kali. Bagian insialisasi umumnya digunakan untuk memberi nilai awal sebuah variable.
- ▶ Sedangkan **terminasi** adalah instruksi yang dilakukan setelah pengulangan selesai dilaksanakan.
- ▶ Ada beberapa bentuk pengulangan yang dapat digunakan, masing-masing dengan syarat dan karakteristik tersendiri.
- ▶ Beberapa bentuk dapat dipakai untuk kasus yang sama, namun ada bentuk yang hanya cocok untuk kasus tertentu saja. Pemilihan bentuk pengulangan untuk masalah tertentu dapat mempengaruhi kebenaran algoritma. Pemilihan bentuk pengulangan yang tepat bergantung pada masalah yang akan diprogram

Struktur pengulangan dengan For

- Pengulangan dengan menggunakan For, merupakan salah teknik pengulangan yang paling tua dalam bahasa pemrograman. Hampir semua bahasa pemrograman menyediakan metode ini, meskipun sintaksnya mungkin berbeda. Pada struktur For kita harus tahu terlebih dahulu seberapa banyak badan loop akan diulang. Struktur ini menggunakan sebuah variable yang biasa disebut sebagai loop's counter, yang nilainya akan naik atau turun selama proses pengulangan.
- Flowchart umum untuk struktur For



Gambar Struktur Algoritma Pengulangan dengan For

- ▶ Dalam mengeksekusi sebuah pengulangan dengan For, urutan langkahlangkah adalah sebagai berikut :
- 1. Menetapkan nilai counter sama dengan awal
- 2. Memeriksa apakah nilai counter lebih besar daripada nilai akhir. Jika benar maka keluar dari proses pengulangan. Apabila kenaikan bernilai negatif, maka proses akan memeriksa apakah nilai counter lebih kecil daripada nilai akhir. Jika benar maka keluar dari proses pengulangan.
- 3. Mengeksekusi pernyataan yang ada di badan loop
- 4. Menaikkan/menurunkan nilai counter sesuai dengan jumlah yang ditentukan pada argument increment. Apabila argument increment tidak ditetapkan maka secara default nilai counter akan dinaikkan 1.
- 5. Ulang kembali mulai langkah no 2.

- ▶ Satu hal yang penting yang harus kita perhatikan adalah nilai counter selalu ditetapkan diawal dari pengulangan.
- ▶ Apabila kita mencoba merubah nilai akhir pada badan loop, maka tidak akan berdampak pada berapa banyak pengulangan akan dilakukan.
- ▶ Algoritma mencetak “Saya akan rajin belajar” sebanyak 5 kali

Flowchart mencetak
“Saya akan rajin belajar”
Sebanyak 5 kali.

Increment = kenaikan + ...
Decrement = penurunan - ...

Simbol Preparation
(Pemberian nilai Awal
Sebuah variabel)



Note :
Pada contoh ini, kita memerlukan variabel counter.
Misalkan kita beri nama i. nilai awalnya adalah 1,
Dan nilai akhirnya adalah 5.
Sedangkan increment atau kenaikan tiap
kali pengulangan dari i adalah satu.
Perintah untuk mencetak pernyataan akan
diulang satu persatu sampai nilai akhir
dari counter terpenuhi (5).

► Bagaimana flowchart untuk mencetak 1,3,5,7,9