

MODUL 7 - 8

PENGULANGAN

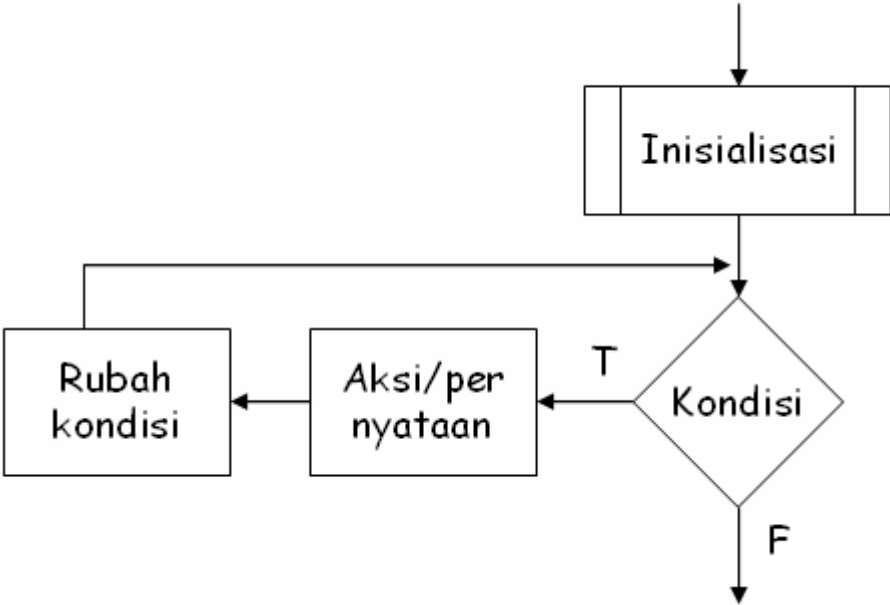
<p>Struktur pengulangan terdiri atas :</p> <ul style="list-style-type: none"> • Kondisi pengulangan : ekspresi boolean • Badan pengulangan • Inisialisasi • Terminasi 	<p>Instruksi (pernyataan) pengulangan</p> <ol style="list-style-type: none"> 1. for 2. while 3. repeat (do - while)
---	--

1. Pernyataan for

Digunakan jika jumlah pengulangan sudah diketahui sebelum eksekusi

Bentuk :

<p>a. for menaik : <u>for</u> <i>pencacah</i> ← <i>nilai_awal</i> <u>to</u> <i>nilai_akhir</i> <u>do</u> <i>Pernyataan</i> <u>endfor</u></p> <p>b. for menurun : <u>for</u> <i>pencacah</i> ← <i>nilai_akhir</i> <u>downto</u> <i>nilai_awal</i> <u>do</u> <i>Pernyataan</i> <u>endfor</u></p>	<ul style="list-style-type: none"> • <i>pencacah</i> harus bertipe integer atau karakter • <i>pernyataan</i> adalah satu atau lebih instruksi yang diulang • <i>nilai_awal</i> lebih kecil atau sama dengan <i>nilai_akhir</i> • jumlah pengulangan = $nilai_akhir - nilai_awal + 1$
--	---

Flow chart	Struktur C++
 <pre> graph TD Start(()) --> Inisialisasi[Inisialisasi] Inisialisasi --> Kondisi{Kondisi} Kondisi -- T --> Aksi[Aksi/pernyataan] Aksi --> Rubah[Rubah kondisi] Rubah --> Kondisi Kondisi -- F --> End(()) </pre>	<pre>for(init;kondisi;rubah_kondisi) { pernyataan; pernyataan; }</pre>
	<pre>for(init;kondisi;) { pernyataan; pernyataan; rubah_kondisi; }</pre>
	<pre>init; for(;kondisi;) { pernyataan; pernyataan; rubah_kondisi; }</pre>
	<p>rubah_kondisi dapat diletakan diawal, ditengah, atau diakhir badan pengulangan</p>

<p><u>ALGORITMA</u> for</p> <p><u>DEKLARASI</u> I : <u>integer</u></p> <p><u>DESKRIPSI</u> <u>for</u> i←1 to 5 <u>do</u> <u>write</u>(I) <u>endfor</u></p>	<pre> graph TD START([START]) --> I1[I=1] I1 --> Cond{I <= 5} Cond -- T --> Print{{Cetak I}} Print --> Inc[I=I+1] Inc --> Cond Cond -- F --> END([END]) </pre>	<pre>#include<iostream.h> void main(void) { for(int I=1 ; I<= 5 ; I++) cout<<I<<endl; }</pre> <p>Apa keluaran dari program diatas?</p>
--	---	---

Analisis perkembangan nilai I

nilai I	Kondisi $I \leq 5$	Tercetak	Nilai I baru (Setelah $I=I+1$)
1	T	1	2
2	T	2	3
3	T	3	4
4	T	4	5
5	T	5	6
6	F	Keluar dari Loop	

Tentukan output dari program berikut :

<pre>#include<iostream.h> void main(void) { for(int i=1; i<5;i++) cout<<i<<endl; }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=1; i<=10;i+=2) cout<<i<<endl; }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=1; i<=10;i+=2); cout<<i<<endl; }</pre>
<pre>#include<iostream.h> void main(void) { int i=0; for(; i<=10;) { i++; cout<<i<<endl; } }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=1; i<=5;) { cout<<i+1<<endl; i++; } }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=1; i<=5;i++) for(int j=1;j<=i;j++) cout<<"*"; cout<<endl; }</pre>
<pre>#include<iostream.h> void main(void) { for(int i=1; i<5;i+=10) cout<<i<<endl; }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=1; i<5;i+=4) cout<<i<<endl; }</pre>	<pre>#include<iostream.h> void main(void) { for(int i=11; i<=10;i+=2) cout<<i<<endl; }</pre>

Soal : Buat program untuk menghitung rata-rata dari sepuluh buah bilangan, yg diinput satu per satu dari kb.

2. Pernyataan while

Bentuk :

while *kondisi* do
 pernyataan
endwhile

Keterangan :

- ✓ *pernyataan* dilaksanakan berulang kali selama *kondisi* bernilai *true*
- ✓ Pengulangan berhenti jika *kondisi* bernilai *false*
- ✓ Agar *kondisi* berubah ke *false*, dalam **badan pengulangan** harus ada instruksi yang merubah nilai *kondisi*

Flow chart	Struktur C++
<pre> graph TD Start(()) --> Inisialisasi[Inisialisasi] Inisialisasi --> Kondisi{Kondisi} Kondisi -- T --> Aksi[Aksi/pernyataan] Aksi --> Rubah[Rubah kondisi] Rubah --> Kondisi Kondisi -- F --> Exit(()) </pre>	<pre> init; while(kondisi) { pernyataan; pernyataan; rubah_kondisi; } </pre> <p>rubah_kondisi dapat diletakan diawal, ditengah, atau diakhir badan pengulangan</p>

ALGORITMA while

DEKLARASI

I : integer

DESKRIPSI

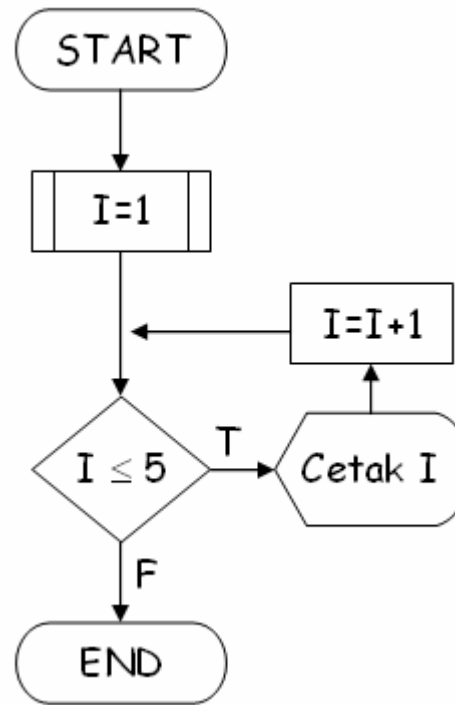
I ← 1

while I ≤ 5 do

write(I)

I ← I + 1

endwhile



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

```
void main(void)
```

```
{
```

```
    int I=1;
```

```
    while(I ≤ 5 )
```

```
    {
```

```
        cout<<I<<endl;
```

```
        I++;
```

```
    }
```

```
}
```

Apa keluaran dari program diatas?

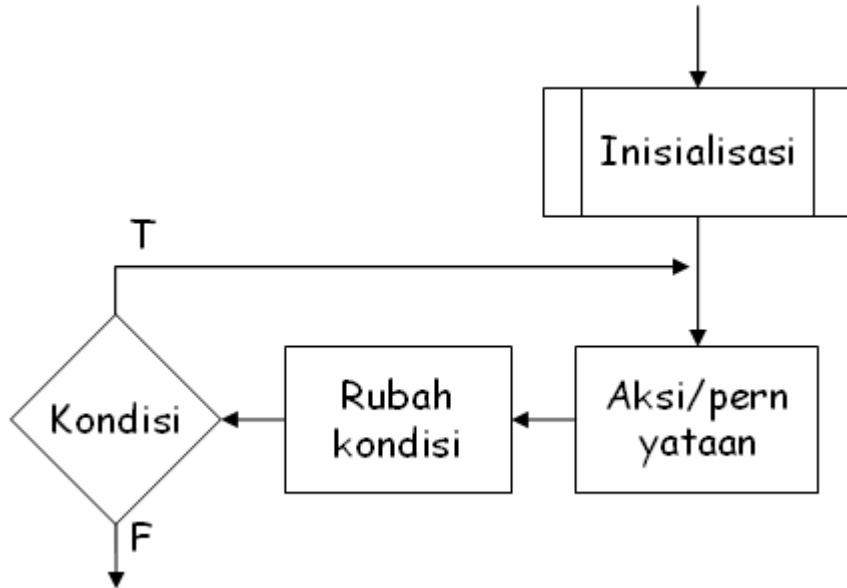
nilai I	Kondisi I ≤ 5	Tercetak	Nilai I baru (Setelah I=I+1)
1	T	1	2
2	T	2	3
3	T	3	4
4	T	4	5
5	T	5	6
6	F	Keluar dari Loop	

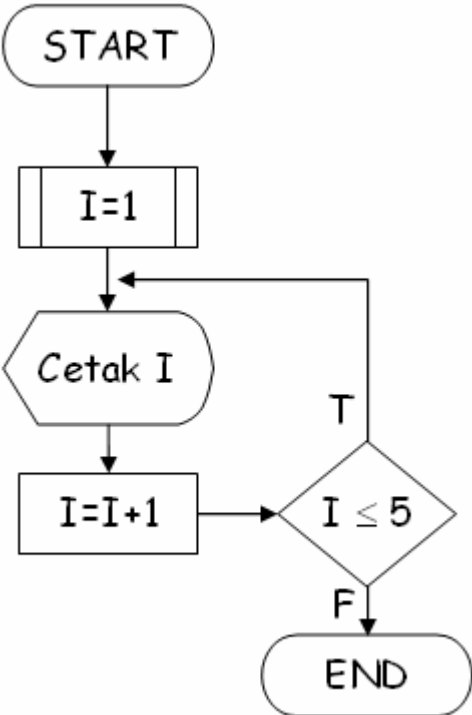
<pre>#include<iostream.h> void main(void) { int i=1; while(i<5) { cout<<i<<endl; i++; } }</pre>	<pre>#include<iostream.h> void main(void) { int i=1; while(i<5) cout<<i++<<endl; }</pre>	<pre>#include<iostream.h> void main(void) { int i=1; while(i<=5) cout<<++i<<endl; }</pre>
<pre>#include<iostream.h> void main(void) { int i=1; while(i=5) { cout<<i<<endl; i++; } }</pre>	<pre>#include<iostream.h> void main(void) { int i=1; while(i) { cout<<i<<endl; i++; } }</pre>	<pre>#include<iostream.h> void main(void) { int i=1; while(0) { cout<<i<<endl; i++; } }</pre>

Soal : Ada sejumlah bilangan integer nilai ujian mahasiswa. Jumlahnya tidak diketahui. Yang diketahui adalah bilangan yang nilainya 999, suatu bilangan yang dipastikan bukan merupakan nilai ujian. Susun program untuk menginput bilangan tsb dan menghitung serta mencetak nilai rata-rata dari semua nilai ujian.

3. Pernyataan repeat(do-while)

<p>Bentuk :</p> <p><u>repeat</u></p> <p> Pernyataan</p> <p><u>until</u> kondisi</p>	<p>Keterangan:</p> <ul style="list-style-type: none"> ✓ Pengulangan dilakukan terus menerus jika <i>kondisi</i> masih <i>true</i> ✓ Pengulangan berhenti jika nilai <i>kondisi</i> menjadi <i>false</i> ✓ Agar <i>kondisi</i> berubah ke <i>false</i>, dalam badan pengulangan harus ada instruksi yang merubah nilai <i>kondisi</i>
--	--

Flow chart	Struktur C++
 <pre> graph TD Start(()) --> Inisialisasi[Inisialisasi] Inisialisasi --> Aksi[Aksi/ Pernyataan] Aksi --> Rubah[Rubah kondisi] Rubah --> Kondisi{Kondisi} Kondisi -- T --> Aksi Kondisi -- F --> Exit(()) </pre>	<pre> init; do { pernyataan; pernyataan; rubah_kondisi; } while(kondisi) </pre> <p>rubah_kondisi dapat diletakan diawal, ditengah, atau diakhir badan pengulangan</p>

<p>ALGORITMA repeat</p> <p>DEKLARASI I : <u>integer</u></p> <p>DESKRIPSI I ← 1 <u>repeat</u> <u>write</u>(I) I ← I+1 <u>until</u> (I ≤ 5)</p>	 <pre> graph TD START([START]) --> I1[I=1] I1 --> CetakI{{Cetak I}} CetakI --> Iplus[I=I+1] Iplus --> Cond{I ≤ 5} Cond -- T --> CetakI Cond -- F --> END([END]) </pre>	<pre> #include<iostream.h> void main(void) { int I=1; do { cout<<I<<endl; I++; } while(I<= 5); } </pre> <p>Apa keluaran dari program diatas?</p>
--	--	--

nilai I	Tercetak	Nilai I baru (Setelah I=I+1)	Kondisi I ≤ 5
1	1	2	T
2	2	3	T
3	3	4	T
4	4	5	T
5	5	6	F, keluar dari loop

Pertanyaan : Apa perbedaan dan persamaan penggunaan pernyataan for, while, do-while?

4. Pernyataan continue

Pernyataan continue dimaksudkan untuk memaksa melanjutkan loop (for, while, do-while)

Contoh untuk pernyataan **while** :

<pre> graph TD START([START]) --> I0[I=0] I0 --> Cond1{I <= 5} Cond1 -- T --> Cond2{I == 3} Cond1 -- F --> END([END]) Cond2 -- T --> Iplus[I=I+1] Cond2 -- F --> PrintI[/Cetak I/] Iplus --> Cond1 PrintI --> Iplus </pre>	<pre> #include<iostream.h> void main(void) { int i=0; while(i<5) { if(i==3) { i++; continue; } cout<<i<<endl; i++; } } </pre>
<p>Pada saat i=3 maka i dinaikan 1 dan kondisi while diuji lagi , akibatnya pernyataan : cout<<i<<endl; i++; tidak dijalankan saat i sama dengan 3</p>	<p>Hasilnya dilayar akan tercetak :</p> <pre> 0 1 2 4 </pre>

Contoh untuk pernyataan **for** :

<pre> graph TD START([START]) --> I0[I=0] I0 --> D1{I ≤ 5} D1 -- T --> D2{I == 3} D1 -- F --> END([END]) D2 -- F --> Iplus[I = I + 1] Iplus --> D1 D2 -- T --> CetakI[/Cetak I/] CetakI --> D2 </pre>	<pre> #include<iostream.h> void main(void) { int i; for(i=0;i<=5;i++) { if(i==3) continue; cout<<i<<endl; } } </pre>
<p>Pada saat i=3 maka i dinaikan 1 dan loop for diulangi lagi , akibatnya pernyataan :</p> <pre> cout<<i<<endl; i++; </pre> <p>tidak dijalankan saat i sama dengan 3</p>	<p>Hasilnya dilayar akan tercetak :</p> <pre> 0 1 2 4 </pre>

5. Pernyataan break

Pernyataan break dimaksudkan untuk memaksa keluar dari loop (for, while, do-while)

Contoh :

<pre> graph TD START([START]) --> I0[I=0] I0 --> D1{I <= 5} D1 -- T --> D2{I == 3} D1 -- F --> END([END]) D2 -- T --> Iplus[I=I+1] Iplus --> D1 D2 -- F --> PrintI[/Cetak I/] PrintI --> Iplus </pre>	<pre> #include<iostream.h> void main(void) { int i; for(i=0;i<=5;i++) { if(i==3) break; cout<<i<<endl; } } </pre>
<p>Pada saat i=3 maka eksekusi langsung keluar dari loop , akibatnya pernyataan :</p> <pre> cout<<i<<endl; i++; </pre> <p>tidak dijalankan saat i sama dengan 3</p>	<p>Hasilnya dilayar akan tercetak :</p> <pre> 0 1 2 </pre>

Soal :

1. Susunlah algoritma dan program C++ untuk menginput sebuah bilangan bulat positif maksimum 255 yang menyatakan suatu bilangan desimal. Kemudian konversikan dan cetak kedalam bentuk biner.
2. Buat algoritma dan program C++ untuk menghitung dan mencetak 10 suku pertama deret bilangan berikut :
1,2,4,8,16,.....
3. Sebuah roket ditembakkan tegak lurus keatas. Pada kilometer pertama menghabiskan 10 kg bahan bakar. Pada kilometer ke-2 menghabiskan 8 kg bahan bakar (atau 80% dari kilometer sebelumnya). Demikian seterusnya. Susun algoritma dan program C++ untuk menghitung dan mencetak total bahan bakar yang dihabiskan setelah mencapai 100 km.