



# **TIB21 – PEMROGRAMAN DASAR**

U N I V E R S I T A S   B U N D A   M U L I A



# **RUNTUNAN DALAM BAHASA PEMROGRAMAN C++**

**Pertemuan ke-3 dan 4**

# Sub-CPMK

Mahasiswa mampu mengaplikasikan program control bahasa pemrograman C++ untuk kasus runtunan (C3, A3)

## Materi

Struktur kontrol runtunan dalam bahasa C++

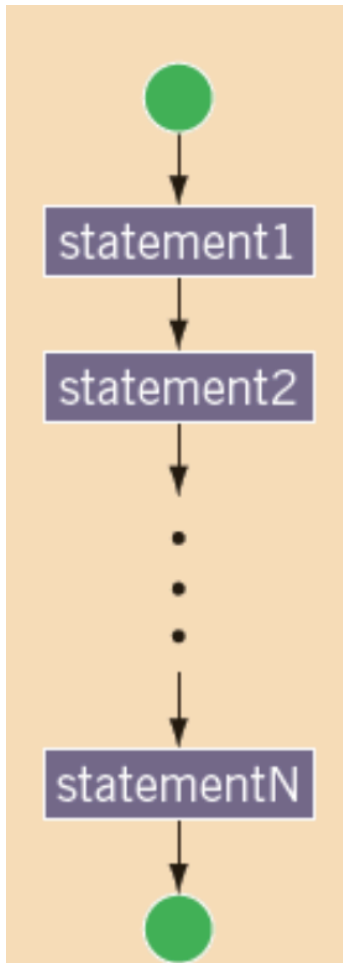
# Referensi

1. Referensi 1, Bab 4 poin 4.4, hal 106-109
2. Referensi 2, Bab 6, hal 106-109



# **Materi: Struktur kontrol runtunan dalam bahasa C++**

# 1. Konsep Runtunan



Pada dasarnya algoritma merupakan runtunan (*sequence*) satu atau lebih instruksi, yang berarti bahwa:

- Tiap insruksi dikerjakan satu per satu.
- Tiap instruksi dilaksanakan tepat sekali; tidak ada instruksi yang diulang.
- Urutan instruksi yang dilaksanakan pemroses sama dengan urutan instruksi sebagaimana yang tertulis di dalam teks algoritmanya.
- Akhir dari instruksi terakhir merupakan akhir algoritma.



## 2. Contoh Kasus Runtunan

### 2.1 Program Mencari Luas Lingkaran


```
9  //Program mencari luas lingkaran
10 #include <iostream>
11 using namespace std;
12
13 int main(){
14     const float phi = 3.14;
15     float r, L;
16
17     cout << "Masukkan nilai jari-jari = ";
18     cin >> r;
19     L = phi * r * r;
20     cout << "Luas lingkaran adalah " << L << endl;
21     return 0;
22 }
```

## 2. Contoh Kasus Runtunan

### 2.1 Program Mencari Luas Lingkaran (Lanj..)



```
Masukkan nilai jari-jari = 5  
Luas lingkaran adalah 78.5  
Program ended with exit code: 0
```

All Output 



## 2. Contoh Kasus Runtunan (Lanj..)

### 2.2 Program Mencari Harga Barang Setelah Diskon

```
1 //Program mencari harga barang setelah diskon
2 #include <iostream>
3 using namespace std;
4
5 int main(){
6     double harga_barang, total_barang, diskon;
7
8     cout << "Berapa harga barang? ";
9     cin >> harga_barang;
10
11     cout << "Berapa diskon yang diberikan? ";
12     cin >> diskon;
13
14     diskon = diskon/100;
15
16     total_barang=harga_barang-(harga_barang*diskon);
17
18     cout << "Diskon yang diberikan adalah sebesar " << diskon << ", dan harga
19         barang setelah diskon menjadi " << total_barang << endl;
20     return 0;
21 }
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.2 Program Mencari Harga Barang Setelah Diskon (Lanj..)

```
▼ ▶
Berapa harga barang? 100000
Berapa diskon yang diberikan? 5
Diskon yang diberikan adalah sebesar 0.05, dan harga barang setelah
    diskon menjadi 95000
Program ended with exit code: 0

All Output ▾ Filter [ ] [ ] [ ]
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.3 Program Konversi dari Fahrenheit ke Celcius

```
9  #include <iostream>
10 using namespace std;
11
12 int main(){
13     int c, f;
14     cout << "Program Konversi Fahrenheit ke Celcius" << endl;
15     cout << "-----" << endl;
16     cout << "Silahkan masukkan suhu dalam derajat fahrenheit = ";
17     cin >> f;
18     c = static_cast<int>(5.0 / 9 * (f - 32) + 0.5);
19     cout << "Suhu setelah dikonversi ke Celcius adalah = " << c << endl;
20     return 0;
21 }
```



Program Konversi Fahrenheit ke Celcius

-----

Silahkan masukkan suhu dalam derajat fahrenheit = 110

Suhu setelah dikonversi ke Celcius adalah = 43

Program ended with exit code: 0

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.4 Program Menghitung Rumus

```
9 // Program untuk mengetahui nilai x
10 #include <iostream>
11 using namespace std;
12
13 int main()
14 { //DEKLARASI
15     float a, b, c, x;
16
17     //ALGORITMA
18     cout << "Nilai a = ";
19     cin >> a;
20     cout << "Nilai b = ";
21     cin >> b;
22     cout << "Nilai c = ";
23     cin >> c;
24
25     x = (((-b)+(2*(c*c))+(4*a*b)))/(2*c));
26
27     cout << "Nilai x adalah " << x << endl;;
28
29     return 0;
30 }
31
```

```
Nilai a = 3
Nilai b = 3
Nilai c = 3
Nilai x adalah 8.5
Program ended with exit code: 0
```

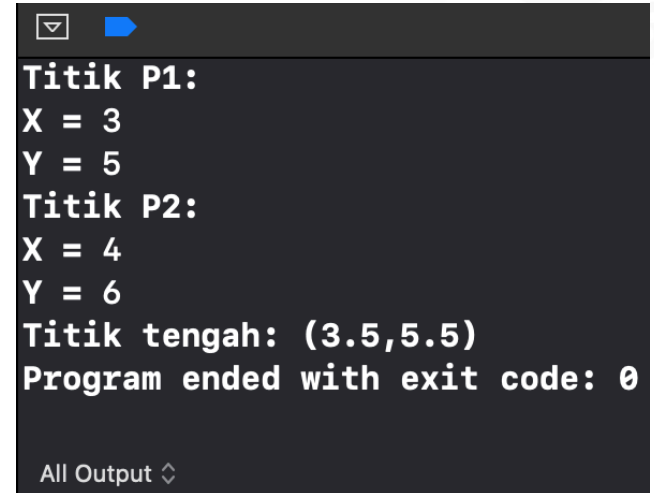
Program menghitung rumus:

$$x = \frac{-b + 2c^2 + 4ab}{2c}$$

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.5 Program Menghitung Titik Tengah

```
9 //PROGRAM HitungTitikTengah
10 #include<iostream>
11 using namespace std;
12
13 //DEKLARASI GLOBAL
14 typedef struct{
15     float x;
16     float y;
17 } Titik;
18
19 Titik P1, P2;
20 Titik Pt;
21
22 int main()
23 {
24     //menerima input dari user
25     cout << "Titik P1: " << endl;
26     cout << "X = ";
27     cin >> P1.x;
28     cout << "Y = ";
29     cin >> P1.y;
30
31     cout << "Titik P2: " << endl;
32     cout << "X = ";
33     cin >> P2.x;
34     cout << "Y = ";
35     cin >> P2.y;
36
37     Pt.x = (P1.x + P2.x)/2;
38     Pt.y = (P1.y + P2.y)/2;
39
40     cout << "Titik tengah: " << "(" << Pt.x << "," << Pt.y << ")" << endl;
41 }
```



```
Titik P1:
X = 3
Y = 5
Titik P2:
X = 4
Y = 6
Titik tengah: (3.5,5.5)
Program ended with exit code: 0

All Output
```

#### Catatan:

Silahkan pahami detail  
pseudocode dari buku Rinaldi  
Munir halaman 100

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.6 Program Konversi Waktu dari Detik

#### Konversi Waktu ke Jam-Menit-Detik

- **TANYA: 4000 detik => jam-menit-detik?**
- $4000 \div 3600 = 1$  (jam)      //1 jam memiliki 3600 detik
- $4000 \bmod 3600 = 400$  (sisa detik)
- $400 \div 60 = 6$  (menit)
- $400 \bmod 60 = 40$  (detik)
- **JAWAB: 4000 detik = 1 jam 6 menit 40 detik**

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.6 Program Konversi Waktu dari Detik (Lanj..)

```
PROGRAM KonversiWaktu
{..}

DEKLARASI:
type Jam: record<hh: integer,
                mm: integer,
                ss: integer
                >

J : Jam           //tipe bentukan
TotalDetik : integer //tipe dasar
Sisa : integer

ALGORITMA:
read(TotalDetik)

J.hh <- TotalDetik div 3600
Sisa <- TotalDetik mod 3600
J.mm <- Sisa div 60
J.ss <- Sisa mod 60

write(J.hh, J.mm, J.ss)
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.6 Program Konversi Waktu dari Detik (Lanj..)

```
9 //Program KonversiWaktu untuk mengkonversi detik ke jam, menit dan detik.
10
11 #include<iostream>
12 using namespace std;
13
14 int main()
15 {
16     //DEKLARASI
17     typedef struct{ int hh;
18                     int mm;
19                     int ss;
20                     } Jam;
21
22     Jam J;
23     int TotalDetik;
24     int Sisa;
25
26     //ALGORITMA
27     cout << "Berapa jumlah detik? ";
28     cin >> TotalDetik;
29
30     J.hh = TotalDetik/3600;
31     Sisa = TotalDetik%3600;
32     J.mm = Sisa/60;
33     J.ss = Sisa%60;
34
35     cout << "Jadi, " << TotalDetik << " detik adalah sama dengan " << J.hh << " jam, " << J.mm << " menit,
36         dan " << J.ss << " detik." << endl;
37
38     return 0;
39 }
```

▼ ▶

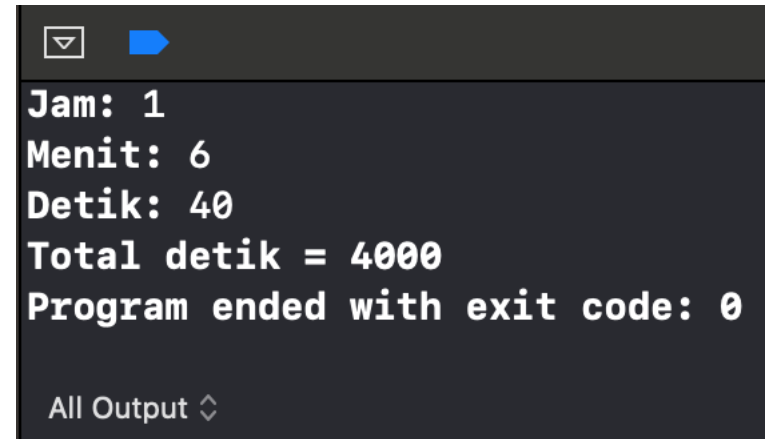
Berapa jumlah detik? 4000  
Jadi, 4000 detik adalah sama dengan 1 jam, 6 menit, dan 40 detik.  
Program ended with exit code: 0



## 2. Contoh Kasus Runtunan (Lanj..)

### 2.7 Program Konversi Waktu ke Detik

```
9 //Program konversi waktu ke detik
10 #include<iostream>
11 using namespace std;
12
13 int main()
14 {
15     //DEKLARASI
16     typedef struct{ int hh;
17                     int mm;
18                     int ss;
19                     } Jam;
20
21     Jam J;
22     int TotalDetik;
23
24     //ALGORITMA
25     cout << "Jam: ";
26     cin >> J.hh;
27
28     cout << "Menit: ";
29     cin >> J.mm;
30
31     cout << "Detik: ";
32     cin >> J.ss;
33
34
35     //sbg output
36     TotalDetik = (J.hh*3600) + (J.mm*60) + J.ss;
37     cout << "Total detik = " << TotalDetik << endl;
38
39     return 0;
40 }
```



```
Jam: 1
Menit: 6
Detik: 40
Total detik = 4000
Program ended with exit code: 0

All Output
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.8 Program Menghitung Durasi Waktu (Lanj..)

#### Panduan:

- JamAwal = 10:00:00 // Detik=  $10 \times 3600 + 0 \times 60 + 0 = 36000$
- JamAkhir = 11:00:00 // Detik=  $11 \times 3600 + 0 \times 60 + 0 = 39600$
- Durasi = 01:00:00 (JamAkhir - JamAwal)

#### Langkah-langkah:

1. Konversi JamAwal ke dalam TotalDetikAwal
2. Konversi JamAkhir ke dalam TotalDetikAkhir
3. Durasi = TotalDetikAkhir - TotalDetikAwal
4. Konversi Durasi (yang dalam detik) ke dalam jam-menit-detik.

#### Clue:

- Langkah 1 dan 2 menggunakan hasil dari program nomor 2.7
- Langkah 4 menggunakan hasil dari program nomor 2.6

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.8 Program Menghitung Durasi Waktu (Lanj..)

```
9 //Program DurasiWaktu
10 #include<iostream>
11 using namespace std;
12
13 int main()
14 {
15     //DEKLARASI
16     typedef struct{ int hh;
17                     int mm;
18                     int ss;
19                     } Jam;
20
21     Jam JamAwal, JamAkhir, DurasiJam;
22     int TotalDetikAwal, TotalDetikAkhir, DurasiDetik;
23     int Sisa;
24
25     //ALGORITMA
26     //Baca Jam Awal
27     cout << "Program Menghitung Durasi Waktu" << endl;
28     cout << "===== " << endl << endl;
29     cout << "Jam Awal: ";
30     cin >> JamAwal.hh;
31
32     cout << "Menit Awal: ";
33     cin >> JamAwal.mm;
34
35     cout << "Detik Awal: ";
36     cin >> JamAwal.ss;
37
38     cout << "-----" << endl;
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.8 Program Menghitung Durasi Waktu (Lanj..)

```
40
41 //Baca Jam Akhir
42 cout << "Jam Akhir: ";
43 cin >> JamAkhir.hh;
44
45 cout << "Menit Akhir: ";
46 cin >> JamAkhir.mm;
47
48 cout << "Detik Akhir: ";
49 cin >> JamAkhir.ss;
50
51
52 //Langkah 1: Konversi JamAwal ke dalam TotalDetikAwal
53 TotalDetikAwal = (JamAwal.hh*3600) + (JamAwal.mm*60) + JamAwal.ss;
54
55 //Langkah 2: Konversi JamAkhir ke dalam TotalDetikAkhir
56 TotalDetikAkhir = (JamAkhir.hh*3600) + (JamAkhir.mm*60) + JamAkhir.ss;
57
58 //Langkah 3: Durasi = TotalDetikAkhir - TotalDetikAwal
59 DurasiDetik = TotalDetikAkhir - TotalDetikAwal;
60 cout << "Durasi = " << DurasiDetik << " detik" << endl << endl;
61
62 //Langkah 4: Konversi Durasi (yang dalam detik) ke dalam jam-menit-detik.
63 DurasiJam.hh = DurasiDetik/3600;
64 Sisa = DurasiDetik%3600;
65 DurasiJam.mm = Sisa/60;
66 DurasiJam.ss = Sisa%60;
67
68 cout << "Jadi, total durasinya adalah " << DurasiJam.hh << " jam, " <<
    DurasiJam.mm << " menit, " << DurasiJam.ss << " detik." << endl;
69
70 return 0;
71 }
```

## 2. Contoh Kasus Runtunan (Lanj..)

### 2.8 Program Menghitung Durasi Waktu (Lanj..)

```
Program Menghitung Durasi Waktu
=====

Jam Awal: 3
Menit Awal: 3
Detik Awal: 3
-----
Jam Akhir: 4
Menit Akhir: 4
Detik Akhir: 4
Durasi = 3661 detik

Jadi, total durasinya adalah 1 jam, 1 menit, 1 detik.
Program ended with exit code: 0

All Output ^
```

### 3. Contoh Penggunaan Beberapa Fungsi

- Manipulator `setw()` digunakan untuk menampilkan keluaran dengan pengaturan kolom atau posisi pada screen.
- Manipulator `setprecision()` digunakan untuk menampilkan digit angka
- Manipulator `string()` digunakan untuk menampilkan keluaran berupa string.
- `Setw()` dan `setprecision()` menggunakan header `<iomanip>`, sedang `string()` menggunakan `<string>`

### 3. Contoh Penggunaan Beberapa Fungsi (Lanj..)

```
9  #include <iostream>
10 #include <iomanip>
11 #include <string>
12 using namespace std;
13
14 int main(){
15     string nama = "Roxy";
16     float nilai = 90.7896543;
17
18     cout << setw(5) << "Nama"
19          << setw(15) << "Nilai" << endl;
20     cout << setprecision(4);
21     cout << setw(5) << nama
22          << setw(15) << nilai << endl;
23
24     return 0;
25 }
26
```

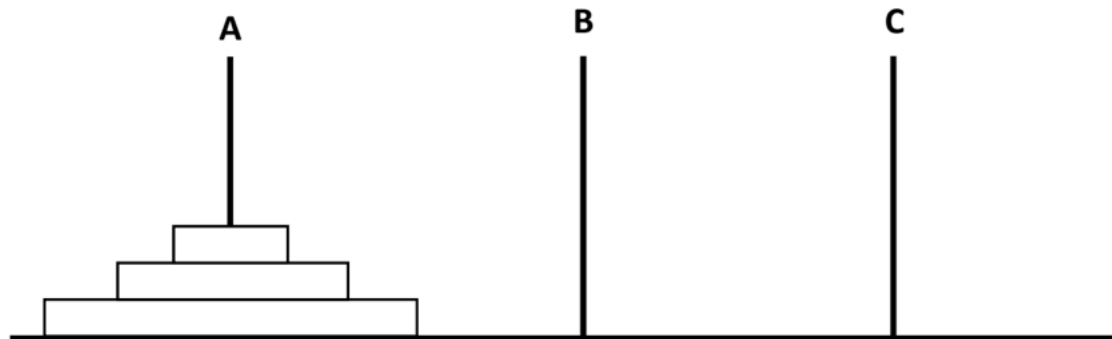
▼ ▶

Nama	Nilai
Roxy	90.79

Program ended with exit code: 0

# Latihan / Review Mandiri

Tiga buah cakram yang masing-masing berdiameter berbeda mempunyai lubang di titik pusatnya. Ketiga cakram tersebut dimasukkan pada sebuah batang besi A sedemikian sehingga cakram yang berdiameter lebih besar selalu terletak di bawah cakram yang berdiameter lebih kecil. Tulislah algoritma dalam notasi program C++ pada batang besi B; setiap kali hanya satu cakram yang boleh dipindahkan, tetapi pada setiap perpindahan tidak boleh ada cakram yang lebih besar berada di atas cakram kecil. Batang besi C dapat dipakai sebagai tempat peralihan dengan tetap memegang aturan yang telah disebutkan.





# Latihan / Review Mandiri (Lanj)

Berapakah nilai A dan B setelah instruksi berikut ini?

DEKLARASI

A, B : integer

ALGORITMA:

$A \leftarrow 10$

$B \leftarrow 2 * A - 5$

$A \leftarrow B$

$A \leftarrow A + 2$

$B \leftarrow A + B$

## Latihan / Review Mandiri (Lanj)

- Latihan soal terkait jam-menit-detik, bisa coba lakukan konversi pseudocode ke C++ pada contoh 6.10 buku Rinaldi Munir, yaitu program untuk menentukan jam berikutnya setelah penambahan satu detik. Misalnya:

Jam Lama	Jam Lama + 1 Detik
22:27:45	22:27:46
22:27:59	22:28:00
22:59:59	23:00:00
23:59:59	00:00:00

# Ringkasan

- Struktur kontrol runtunan digunakan untuk mengeksekusi instruksi secara beruntun (sequence)
- Terdapat beberapa fungsi yang dapat digunakan untuk mengatur layout, seperti `setw()` dan fungsi-fungsi manipulasi lainnya.



# Terima Kasih

U N I V E R S I T A S   B U N D A   M U L I A