

PERTEMUAN 8 CONTROL STATEMENT IF (Lanjutan)

A. Tujuan Pembelajaran

Setelah mempelajari materi ini, mahasiswa mampu:

1. Mengimplementasikan algoritma menggunakan control statement IF dalam pemecahan suatu masalah dengan berbagai alternatif jawaban yang tersedia dengan mengambil keputusan untuk memilih jawaban yang tepat.

Uraian Materi

1.1. Pendahuluan

Dalam seleksi, program mengeksekusi statemen-statemen tertentu tergantung dari kondisi spesifik. Perhatikan tiga statemen berikut:

1. if (skor lebih besar atau sama dengan 90)
 nilai adalah A
2. if (jam kerja kurang dari atau sama dengan 40)
 gaji = gajiPerJam * jam Kerja
 else
 gaji = (gajiPerJam * 40) + (1.5 * (gajiPerJam * jamKerja – 40))
3. if (suhu lebih dari 70 derajat dan tidak hujan)
 mainGolf

Ketiga statemen di atas merupakan contoh kondisional yang akan dieksekusi hanya jika kondisi terpenuhi. Kondisi menjadi terpenuhi jika ia dievaluasi menjadi **true**.

1.2. Jenis-Jenis *Decision If*

1. Perintah *if*

Bentuk umum perintah if:

if (ekspresi) statemen

Ekspresi : Merupakan sebuah kondisi logikal yang digunakan sebagai pembuat keputusan untuk memutuskan apakah mengeksekusi statemen atau tidak.

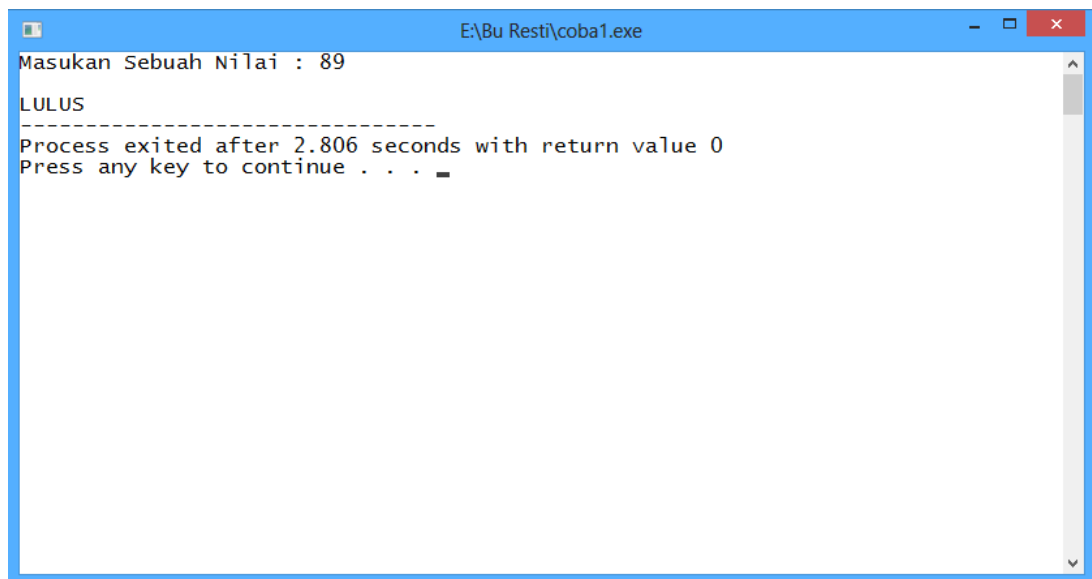
Jika nilai dari **ekspresi** adalah **true**, maka statemen akan dieksekusi dan jika nilai dari **ekspresi** adalah **false** maka statemen tidak akan dijalankan.

Contoh Program 1 :

```
#include<iostream>
using namespace std;
main(){
    int nilai;
    cout<<"Masukan Sebuah Nilai : ";
    cin>>nilai;

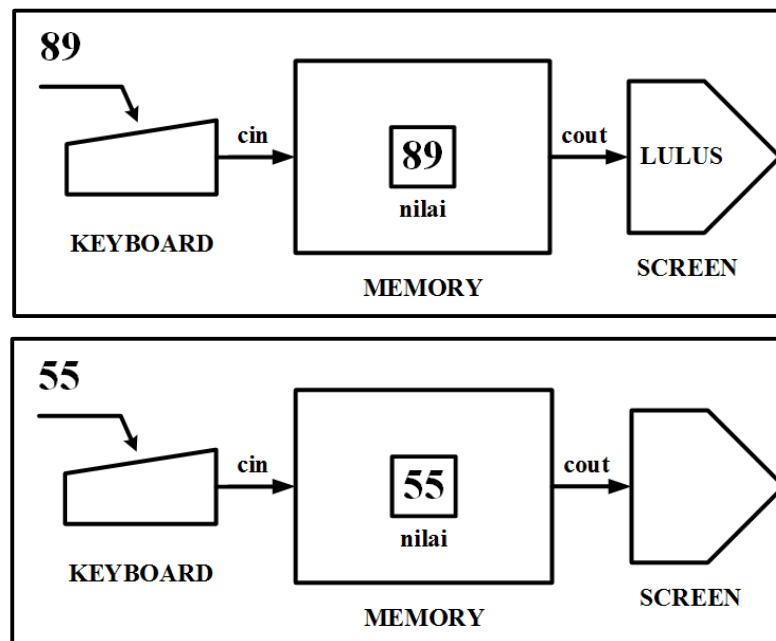
    if(nilai>69)
        cout<<"\nLULUS";
}
```

Output :



```
E:\Bu Resti\coba1.exe
Masukan Sebuah Nilai : 55
-----
Process exited after 2.477 seconds with return value 0
Press any key to continue . . .
```

Penjelasan :



Source code dan output di atas merupakan contoh penggunaan perintah if, dimana ketika nilai yang diinputkan sesuai dengan ekspresi (>69) maka *compiler* akan memproses statemen pertama dan mencetak kata LULUS pada output-an. Tetapi jika nilai tidak sesuai dengan ekspresi (<69) tidak ada kata yang tercetak pada output dikarenakan statemen yang terdapat pada program hanya satu (tidak memiliki statemen bernilai false).

2. Perintah *if ... else*

Bentuk umum perintah if ... else:

```
if (ekspresi)
    statemen1
else
    statemen2
```

Ekspresi : Merupakan sebuah ekspresi logikal yang digunakan sebagai pembuat keputusan untuk memutuskan apakah mengeksekusi statemen atau tidak.

Ada banyak situasi pemrograman dimana di dalamnya harus memilih salah satu dari dua alternatif. Untuk memilih salah satu dari dua alternatif, C++ menyediakan statemen `if ... else` yang diawali dengan kata kunci **if**, diikuti dengan sebuah ekspresi logikal yang diapit oleh sepasang kurung, diikuti dengan sebuah statemen, diikuti dengan katakunci **else**, dan diikuti dengan statemen kedua. Jika nilai dari **ekspresi** adalah **true**, maka **statemen1** akan dieksekusi. Jika nilai dari **ekspresi** adalah **false**, maka **statemen2** yang dieksekusi.

Contoh Program 2 :

```
#include<iostream>
using namespace std;
main(){
    int nilai;
    cout<<"Masukan Sebuah Nilai : ";
    cin>>nilai;

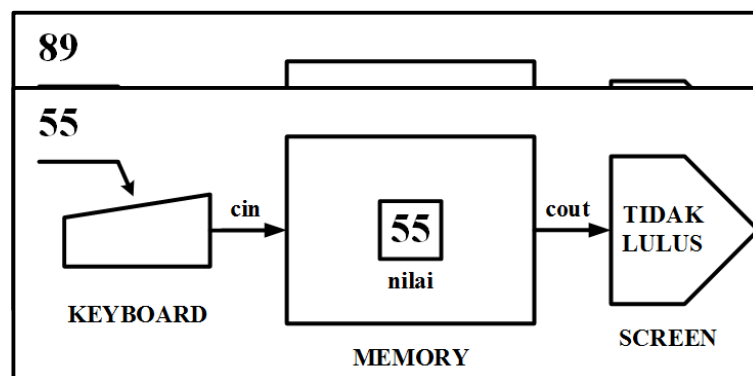
    if(nilai>69)
        cout<<"\nLULUS";
    else
        cout<<"\nTIDAK LULUS"
}
```

Output:

```
E:\Bu Resti\coba1.exe
Masukan Sebuah Nilai : 89
LULUS
-----
Process exited after 0.9294 seconds with return value 0
Press any key to continue . . .
```

```
E:\Bu Resti\coba1.exe
Masukan Sebuah Nilai : 55
TIDAK LULUS
-----
Process exited after 2.48 seconds with return value 0
Press any key to continue . . .
```

Penjelasan:



Source code dan *output* di atas merupakan contoh penggunaan perintah `if ... else`, dimana ketika nilai yang diinputkan sesuai dengan ekspresi (`>69`) maka *compiler* akan memproses statemen pertama dan mencetak kata LULUS pada output-an. Jika nilai tidak sesuai dengan ekspresi (`<69`), *compiler* akan memproses statemen kedua dan mencetak kata TIDAK LULUS (karena pada program terdapat statemen bernilai false).

3. Perintah *if* bersarang (*Nested-If*)

Bentuk umum *if* bersarang:

```
if (ekspresi1)
    if (ekspresi2)
        statemen1
    else
        statemen2
else
    statemen3
```

Merupakan sebuah ekspresi logikal yang digunakan sebagai

Ekspresi1,2 : pembuat keputusan untuk memutuskan apakah mengeksekusi statemen atau tidak.

Jika suatu statemen kendali berada di dalam statemen kendali yang lain, statemen kendali tersebut dikatakan bersarang di dalam statemen kendali yang lain. Jika nilai dari **ekspresi1** adalah **true**, maka proses pengecekan akan dilanjutkan ke **ekspresi2**. Jika nilai **ekspresi2** adalah **true**, maka **statemen1** akan dieksekusi. Jika nilai **ekspresi2** adalah **false**, maka **statemen2** akan dieksekusi. Jika nilai **ekspresi1** adalah **false**, maka **statemen3** akan dieksekusi.

Contoh Program 3:

Susun program dalam bahasa C++ untuk menginput tiga buah bilangan bulat (integer), dimana ketiga buah bilangan tersebut dianggap bernilai tidak sama, kemudian mencetak salah satu bilangan yang nilainya terbesar.

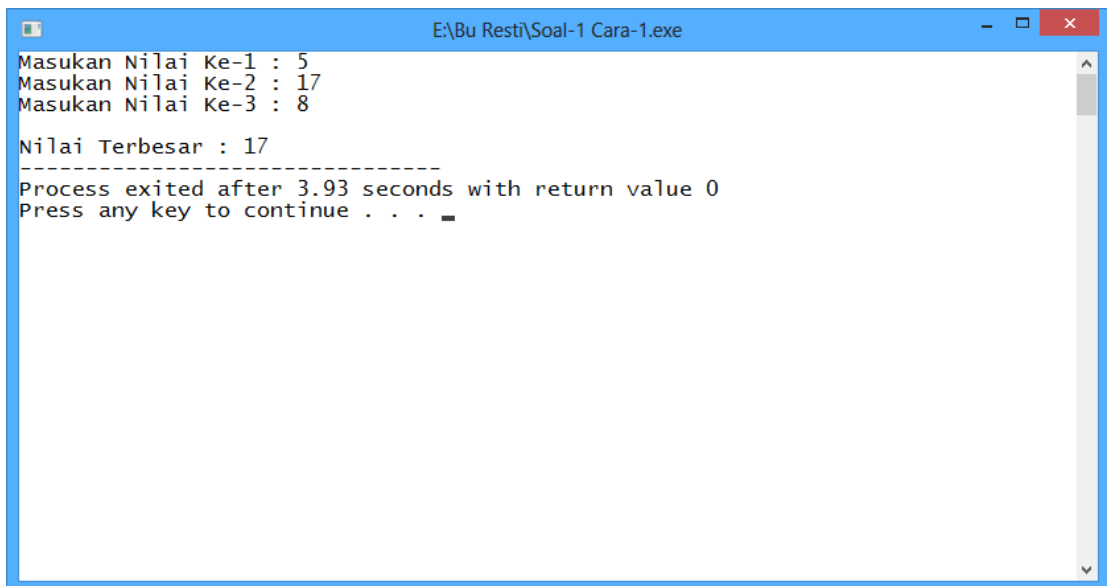
```

#include<iostream>
using namespace std;
main() {
    int a, b, c;
    cout<<"Masukan Nilai Ke-1 : ";
    cin>>a;
    cout<<"Masukan Nilai Ke-2 : ";
    cin>>b;
    cout<<"Masukan Nilai Ke-3 : ";
    cin>>c;

    if(a>b)
        if(a>c)
            cout<<"\nNilai Terbesar
: "<<a;
        else
            cout<<"\nNilai Terbesar
: "<<c;
    else
        if(b>c)
            cout<<"\nNilai Terbesar
: "<<b;
        else
            cout<<"\nNilai Terbesar
: "<<c;
}

```

Ouput:



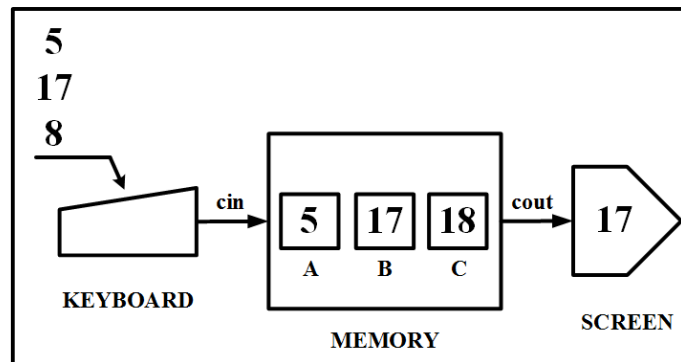
```

E:\Bu Resti\Soal-1 Cara-1.exe
Masukan Nilai Ke-1 : 5
Masukan Nilai Ke-2 : 17
Masukan Nilai Ke-3 : 8

Nilai Terbesar : 17
-----
Process exited after 3.93 seconds with return value 0
Press any key to continue . . .

```

Penjelasan:



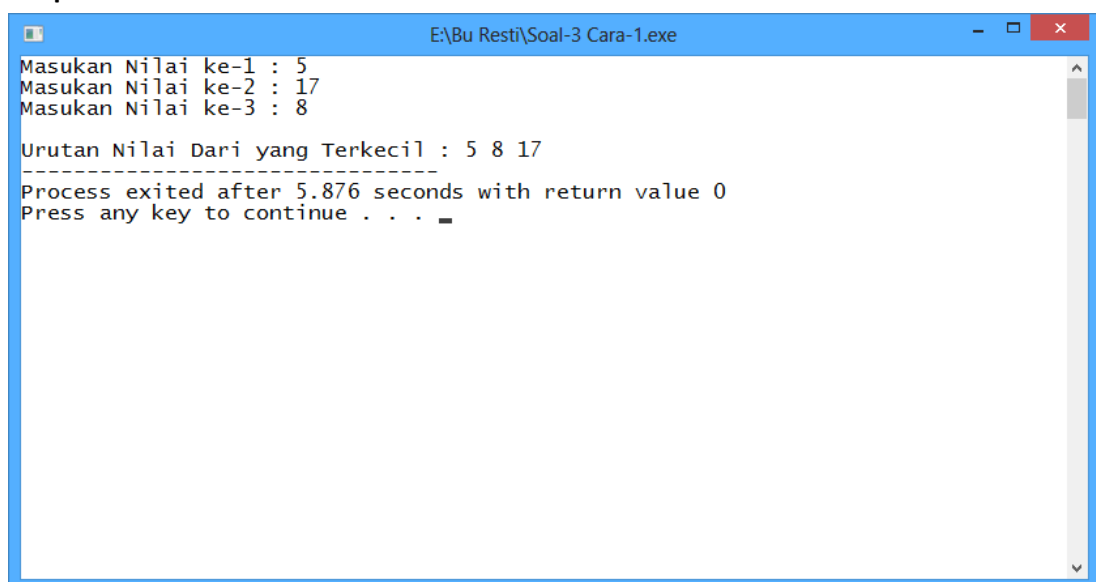
Dari *source code* dan *output* diatas, untuk nilai integer yang diinputkan adalah 5, 17, 8. Berdasarkan program yang telah dibuat untuk menentukan nilai terbesar dari tiga nilai yang diinputkan, *compiler* akan melakukan pengecekan dari ketiga nilai tersebut mana yang nilai terbesar. Setelah proses pengecekan program akan mencetak hasilnya yakni 17, karena 17 merupakan nilai terbesar jika dibandingkan dengan 5 dan 8.

Contoh Program 4:

Susun program untuk menginput tiga buah bilangan bulat (misal a, b dan c dimana $a < b < c > a$), kemudian mencetak ketiga nilai tersebut urut nilai dari terkecil ke besar.

```
#include<iostream>
using namespace std;
main(){
    int a, b, c;
    cout<<"Masukan Nilai ke-1 : "; cin>>a;
    cout<<"Masukan Nilai ke-2 : "; cin>>b;
    cout<<"Masukan Nilai ke-3 : "; cin>>c;
    if(a<b)
        if(b<c)
            cout<<"\nUrutan Nilai Dari yang Terkecil : 
"<<a<<" "<<b<<" "<<c<<" ";
        else
            if(a<c)
                cout<<"\nUrutan Nilai Dari yang Terkecil : 
"<<a<<" "<<c<<" 
                    "<<b<<" ";
            else
                cout<<"\nUrutan Nilai Dari yang Terkecil 
: "<<c<<" "<<a<<" 
                    "<<b<<" ";
```

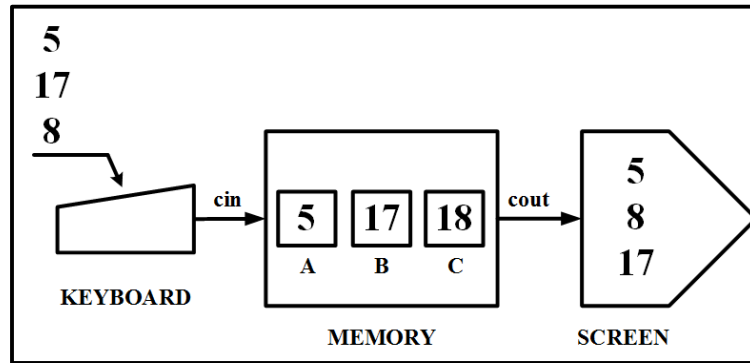

Output



```
E:\Bu Resti\Soal-3 Cara-1.exe
Masukan Nilai ke-1 : 5
Masukan Nilai ke-2 : 17
Masukan Nilai ke-3 : 8

Urutan Nilai Dari yang Terkecil : 5 8 17
-----
Process exited after 5.876 seconds with return value 0
Press any key to continue . . .
```

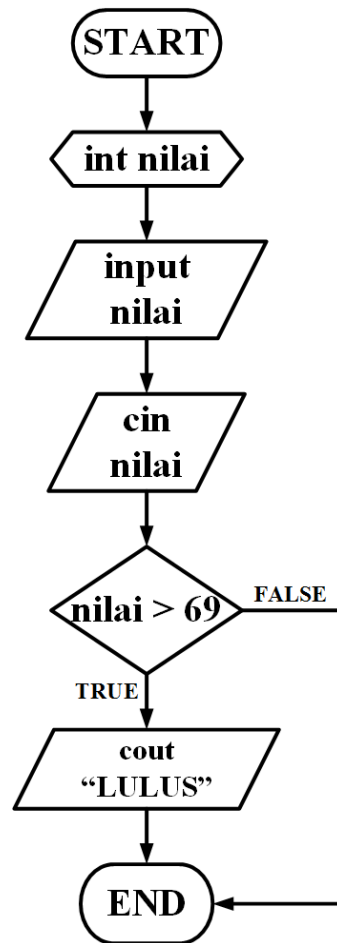
Penjelasan



Berdasarkan *source code* dan output di atas, *nested-if* atau *if* bersarang dapat digunakan dalam pembuatan program untuk mengurutkan nilai dari yang terkecil hingga terbesar. Setelah nilai diinputkan secara acak (sebagai contoh, nilai yang diinputkan adalah 5, 17 dan 8), *compiler* akan melakukan proses pengurutan nilai-nilai tersebut. **Statemen pertama** akan dieksekusi apabila nilai a lebih kecil dari nilai b dan nilai b lebih kecil dari nilai c. **Statemen kedua** akan dieksekusi apabila nilai a lebih kecil dari nilai b, nilai c lebih kecil dari nilai b dan nilai a lebih kecil dari nilai c. **Statemen ketiga** akan dieksekusi apabila nilai a lebih kecil dari nilai b, nilai c lebih kecil dari nilai b dan nilai c lebih kecil dari nilai a. **Statemen keempat** akan dieksekusi apabila nilai b lebih kecil dari nilai a dan nilai a lebih kecil dari nilai c. **Statemen kelima** akan dieksekusi apabila nilai b lebih kecil dari nilai a, nilai c lebih kecil dari nilai a dan nilai b lebih kecil dari nilai c. **Statemen keenam** akan dieksekusi apabila nilai b lebih kecil dari nilai a, nilai c lebih kecil dari nilai a dan nilai c lebih kecil dari nilai b.

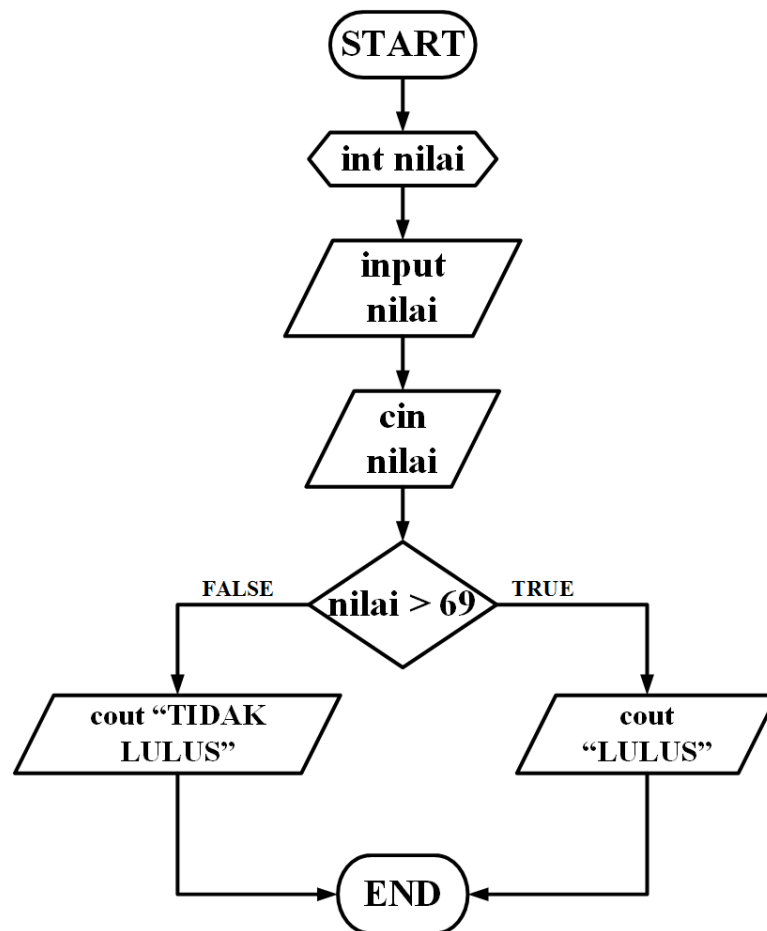
1.3. Alur Kerja Perintah IF

1. Perintah *if*



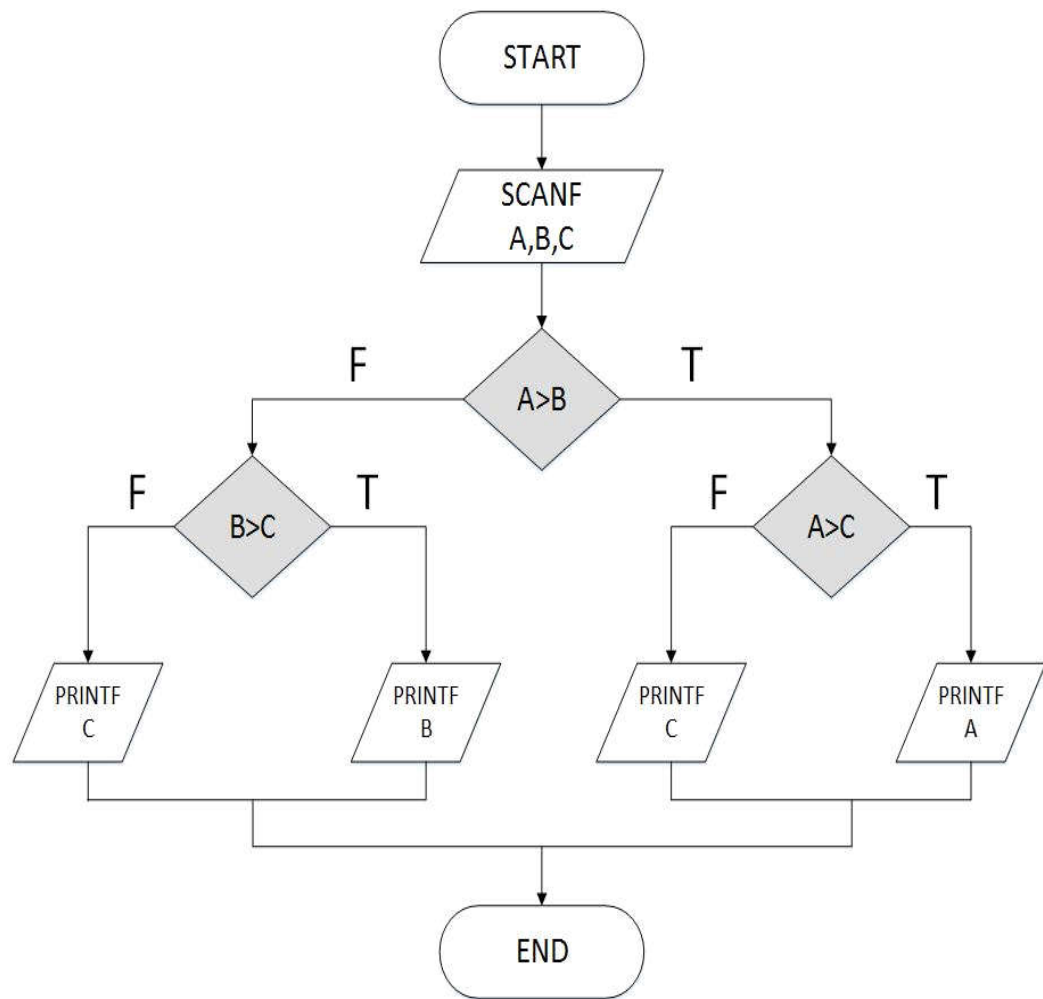
- a. Mulai
- b. Deklarasi varibel nilai
- c. Input nilai
- d. Program membaca inputan nilai
- e. Cek nilai yang diinput
 - 1) Jika nilai lebih besar dari 69, cetak kata "LULUS"
 - 2) Jika nilai lebih kecil dari 69, program berakhir
- f. Selesai.

2. Perintah *if ... else*

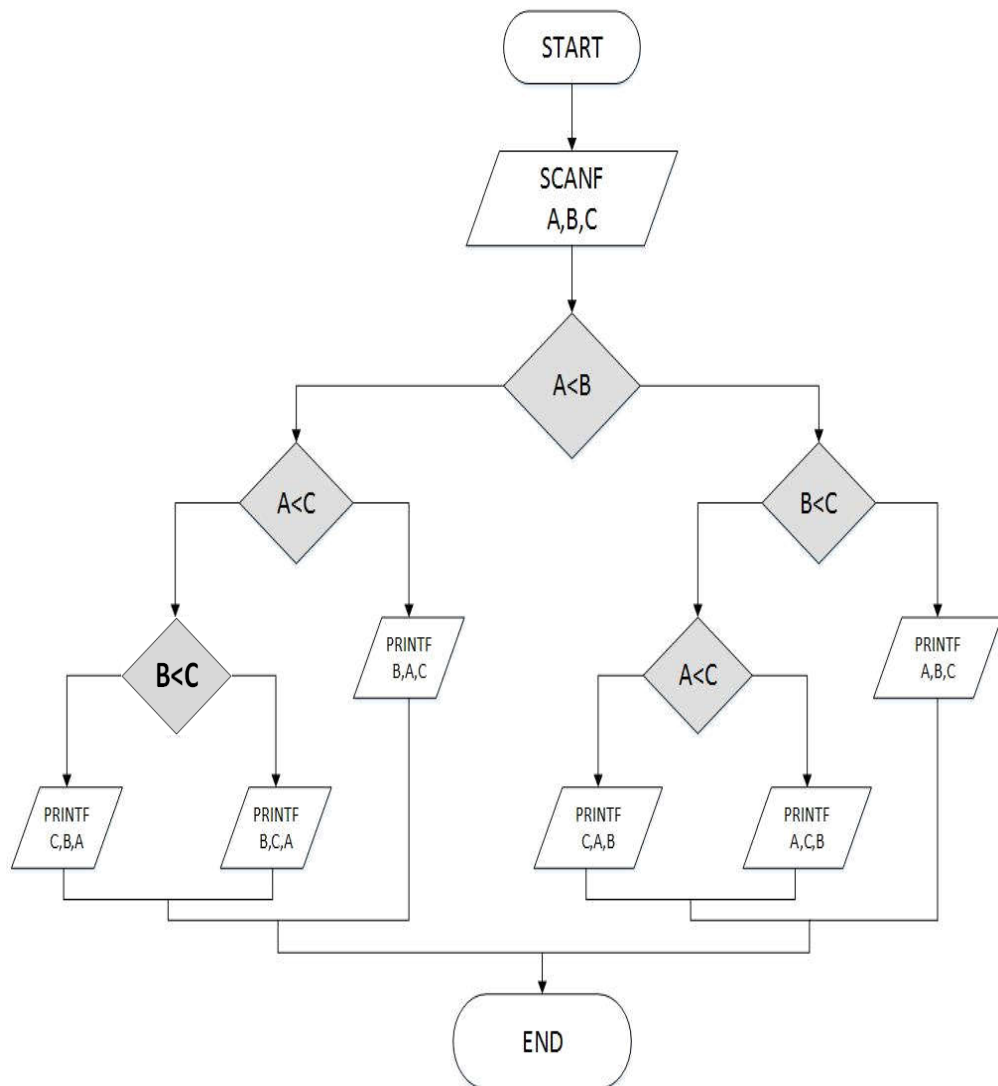


- a. Mulai
- b. Deklarasi variabel nilai
- c. Input nilai
- d. Program membaca inputan nilai
- e. Cek nilai yang diinput
 - 1) Jika nilai lebih besar dari 69, cetak kata "LULUS"
 - 2) Jika nilai lebih kecil dari 69, cetak kata "TIDAK LULUS"
- f. Selesai

3. Perintah *if* bersarang (*nested-if*)



- a. Mulai
- b. Program membaca inputan (nilai A, B, C)
- c. Cek nilai yang diinput
 - 1) Jika A lebih besar dari B dan A lebih besar dari C, cetak A
 - 2) Jika A lebih besar dari B dan A lebih kecil dari C, cetak C
 - 3) Jika A lebih kecil dari B dan B lebih besar dari C, cetak B
 - 4) Jika A lebih kecil dari B dan B lebih kecil dari C, Cetak C
- d. Selesai.



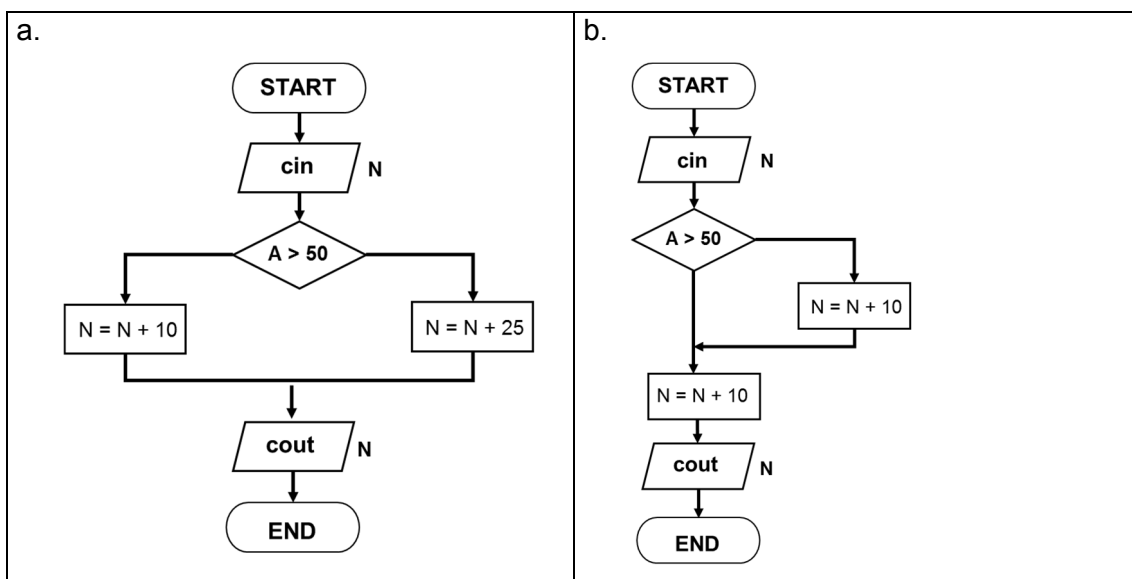
- a. Mulai
- b. Program membaca inputan (nilai A, B, C)
- c. Cek nilai yang diinput
 - 1) Jika A lebih kecil dari B dan B lebih kecil dari C, cetak A, B, C
 - 2) Jika A lebih kecil dari B, B lebih besar dari C, dan A lebih lebih kecil dari C, cetak A, C, B
 - 3) Jika A lebih kecil dari B, B lebih besar dari C, dan A lebih besar dari C, cetak C, A, B
 - 4) Jika A lebih besar dari B dan A lebih kecil dari C, cetak B, A, C
 - 5) Jika A lebih besar dari B, A lebih besar dari C, dan B lebih kecil dari C, cetak B, C, A
 - 6) Jika A lebih besar dari B, A lebih besar dari C, dan B lebih besar dari C, cetak C, B, A
- d. Selesai

Soal/Latihan

1. Gambarkan flowchar dari penggalan program berikut :

<pre> int T=0, A=5, B=10 if(A>B){ T = T+A; T = T+B; cout<<"T : "<<T; }else{ T = T - A; T = T - A; } </pre>	<pre> int T = 50, A = 10, B = 5; if (A<B){ cout<<T; A *= T; cout<<A; }else{ T /= B; cout<<T; } </pre>
---	--

- Susun algoritma dan program atau penggalan program untuk menginput 3 buah bilangan yang masing-masing menyatakan panjang sebuah garis. Kemudian periksa ketiga buah garis tersebut apakah dapat membentuk sebuah segitiga atau tidak. Bila ketiga buah garis tersebut dapat membentuk sebuah segitiga, maka cetak perkataan "SEGI TIGA", sebaliknya bila ketiga garis tersebut tidak mungkin membentuk sebuah segitiga, maka cetak perkataan: "BUKAN SEGITIGA"
- Buat / tulis program dengan bahasa C++ untuk menyatakan algoritma yang di gambarkan flowchar berikut:



- Untuk program dengan algoritma pada soal no 3, Apa yang akan tercetak jika nilai N yang di masukkan:
 - 25
 - 50
 - 75

Referensi

A.S, R. (2018). *LOGIKA ALGORITMA dan PEMROGRAMAN DASAR*. BANDUNG: MODULA.

Davis, S. R. (2014). *C++ For Dummies* (7th ed.). John Wiley & Sons, Inc.

Deitel, P., & Deitel, H. (2014). *C++ How To Program* (9th ed.). United State of America: Pearson.

Kristanto, A. (2003). *Algoritma & Pemograman Dengan C++*. Yogyakarta: Graha Ilmu.

Munir, R. (2005). *Algoritma dan Pemrograman dalam Bahasa Pascal dan C*. Bandung: Penerbit Informatika.

Sjukani, M. (2014). *Algoritma dan Struktur Data 1 dengan C, C++ dan Java* (Edisi 9 ed.). Jakarta: Mitra Wacana Media.