

PERTEMUAN 9 CONTROL STATEMENT MENGGUNAKAN NESTED IF, SWITCH dan LOGICAL OPERATOR

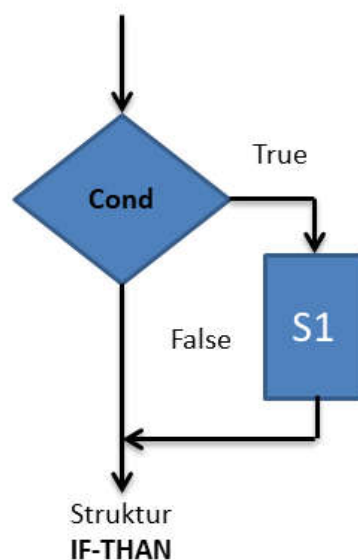
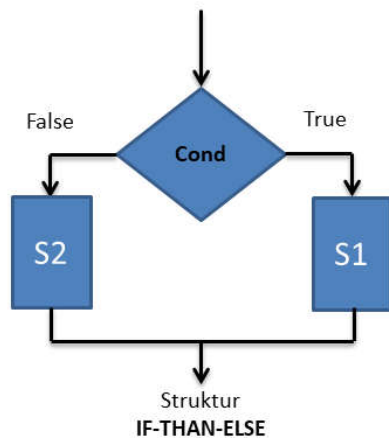
A. Tujuan Pembelajaran

Setelah mempelajari materi ini, mahasiswa mampu memahami tentang penggunaan control statement menggunakan NESTED if dan switch dan Logical Operator dapat menggunakannya dalam pemrograman

MATERI

1.1 Nested if

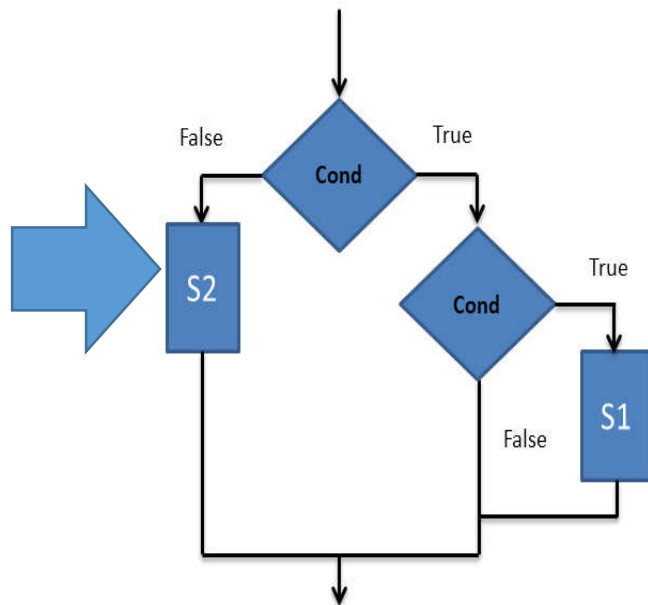
Perhatikan kembali struktur **IF-THEN-ELSE** dan **IF-THEN** statement seperti yang sudah diterangkan melalui flowchart sebagai berikut:



Dari ilustrasi gambar, terlihat **S** adalah satu atau sekelompok statement. Didalam kelompok **S** mungkin terdapat statement **IF** sehingga terjadi IF secara berjenjang atau secara tersarang yang biasa disebut **Nested If** (nest=sarang).

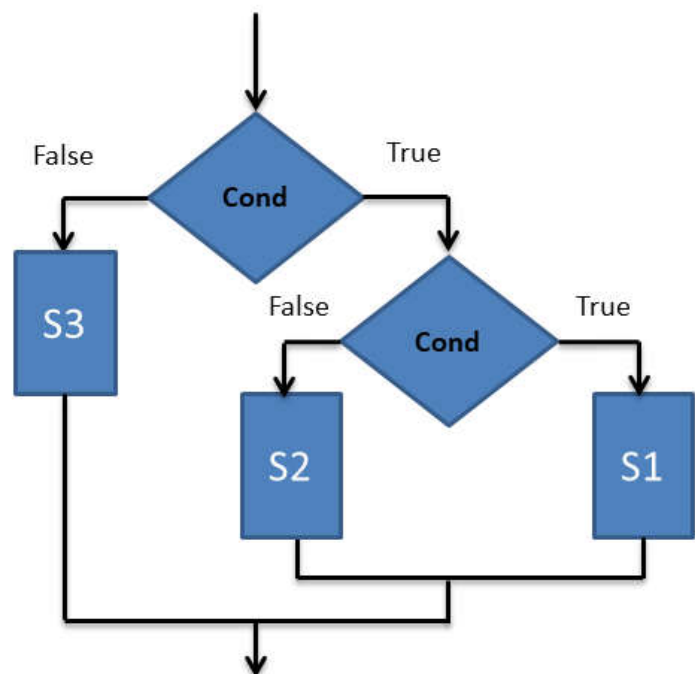
Contoh -1

```
If (cond1)
  {if (cond2)
    {
      S1
    }
  }
else {
  S2
}
```



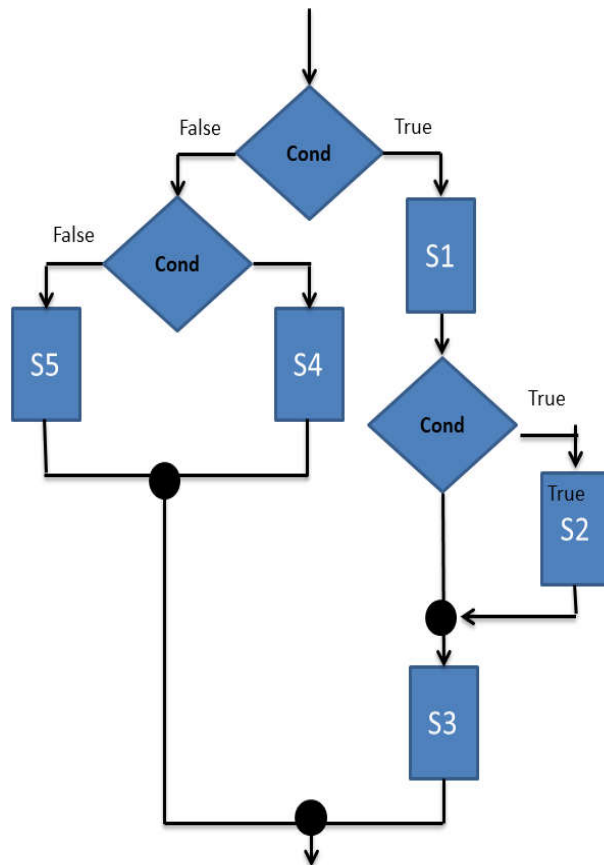
Contoh -2

```
If (cond1)
  {if (cond2)
    {
      S1
    }
  }
else {
  S2
}
else {
  S3
}
```



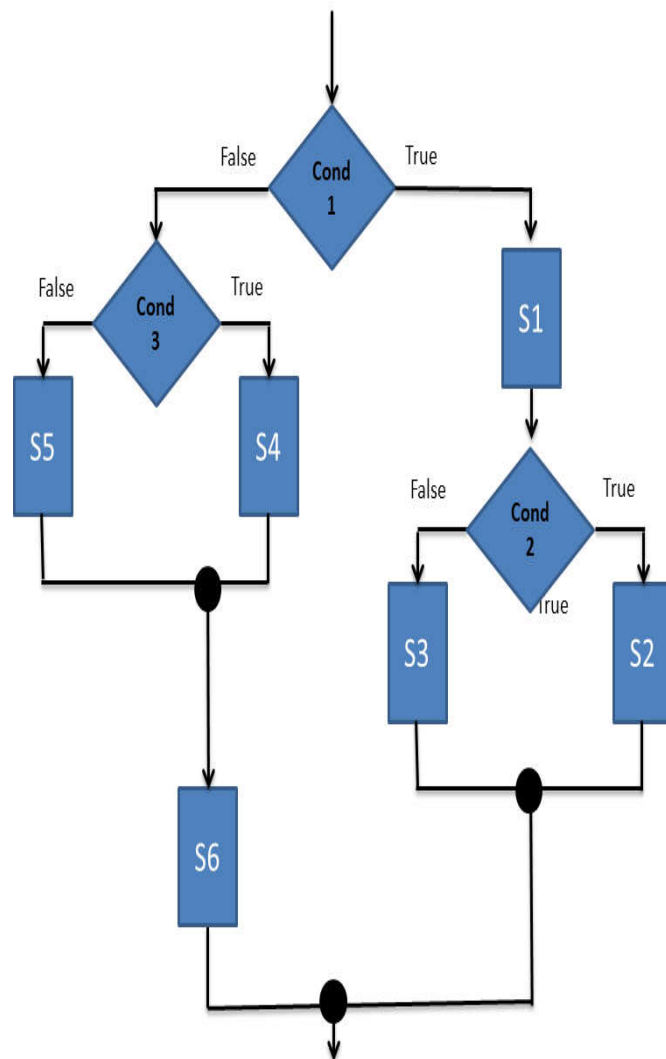
Contoh -3

```
If (cond1)
{
  S1
  if (cond2)
  {
    S2
  }
  S3
}else
{ if (cond3)
{
  S4
}
else {
  S5
}}
}}
```



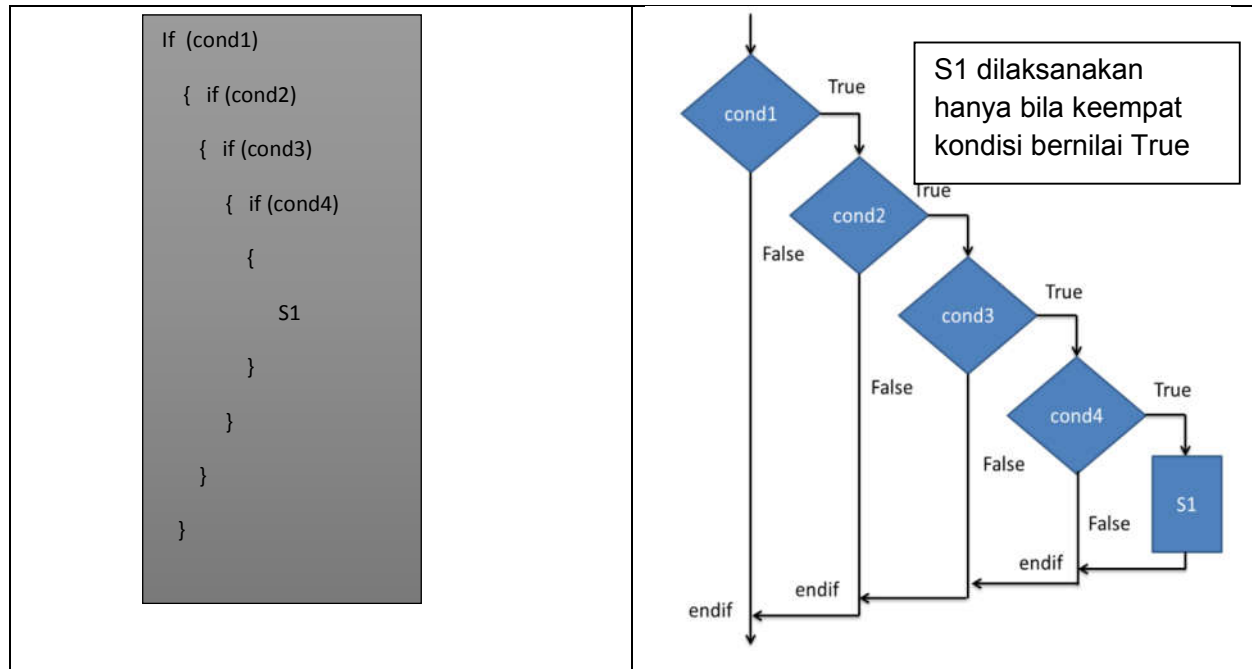
Contoh -4

```
if (cond1)
{
    S1
    if (cond2)
    {
        S2
    }
}
else
{
    S3
}
else
{
    if (cond3)
    {
        S4
    }
    else
    {
        S5
    }
}
S6
}
```

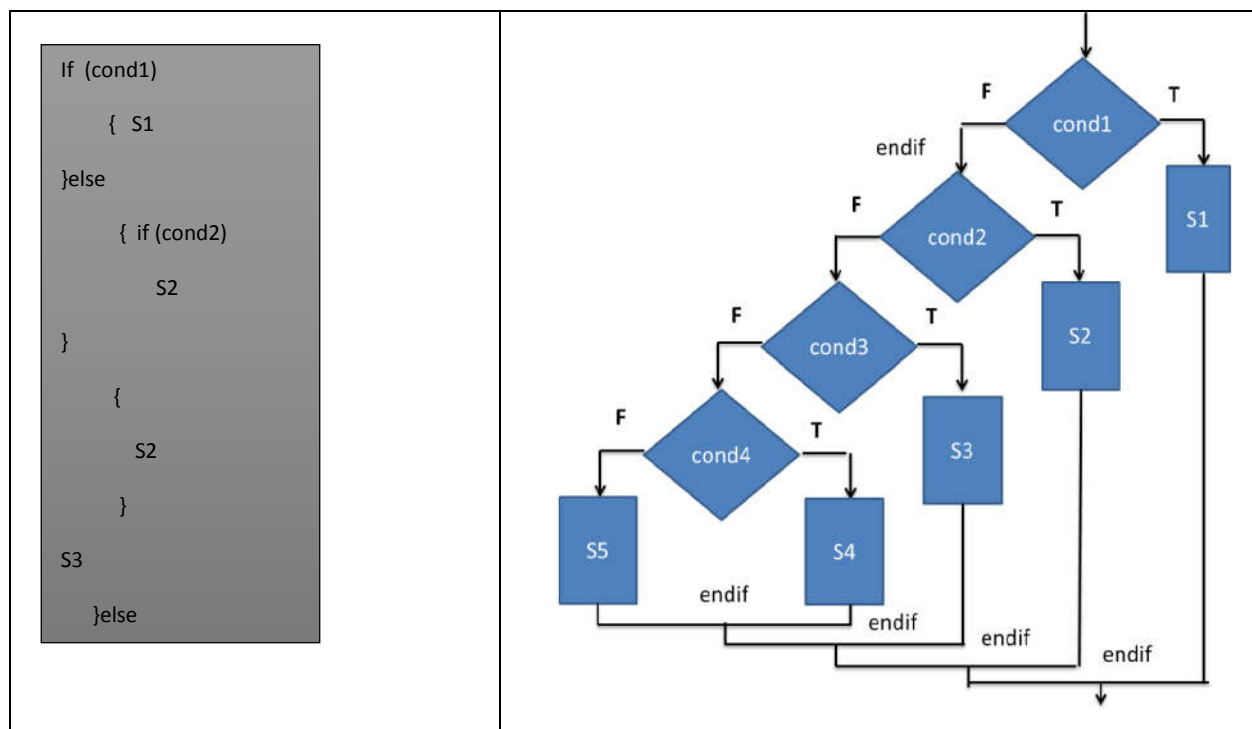


Perhatikan posisi letak Endif dalam flowchart. Posisi ini penting untuk menganalisa aliran terutama untuk nested if yang kompleks atau untuk proses pengulangan yang bersifat rekursif.

Contoh -5



Contoh -6



Struktur pada contoh 5 dan 6 terlihat nested if berjenjang seperti anak tangga sehingga sering juga disebut struktur if-else-if ladder.

1.2 Milti Condition dan Logical Operator

Terkadang satu kondisi saja tidak cukup untuk menentukan suatu syarat sehingga diperlukan dua atau lebih kondisi. Untuk menggabungkan kondisi-kondisi tersebut digunakan operator yang disebut logical operator.

Ada 3 macam logical operator yaitu:

- NOT → !
- AND → &&
- OR → ||

a. Operator Not

Operator not walaupun sifatnya logical tapi beberapa buku literatur tidak memasukkannya kedalam logical operator. Operator **NOT** bukan untuk menggabung 2 buah kondisi, tapi bekerja sebagai pembalik nilai logika **TRUE** menjadi **FALSE** dan sebaliknya sehingga sering disebut **Unary Operator**.

Contoh:

Misal A=6 dan B= 3

Condition	Nilai
A == B	False
A > B	True
A < B	False
A >= B	True
A <= B	False
A != B	True

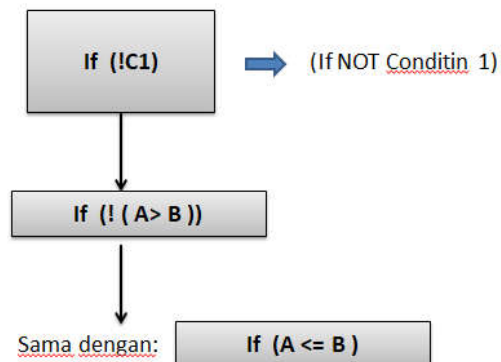


Not(Condition)	Nilai
! (A == B)	True
! (A > B)	False
! (A < B)	True
! (A >= B)	False
! (A <= B)	True
! (A != B)	False

Sehingga

Penulisan	Sama maksudnya
! (A == B)	A != B
! (A > B)	A <= B
! (A < B)	A >= B
! (A >= B)	A < B
! (A <= B)	A > B
! (A != B)	A == B

Catatan:



b. Operator AND

Operator AND menggabung dua buah kondisi menjadi satu nilai sedemikian rupa akan bernilai TRUE hanya bila kedu kondisi digabungkan bernilai TRUE, Atau dengan kata lain, kedua syarat harus terpenuhi.

Pemakaian dalam program C/C++ atau Java

`If (cond1 && cond2)`

Atau

`If (cond1) && (cond2)`

Contoh penulisan dalam bahasa C/C++ ata Java:

```

1) If ( KodeSex == 1 && Umur <=25)
2) If ( Nilai >= 60 && Nilai < 70)
  
```

Perhatikan table yang memperlihatkan nilai gabungan dua buah kondisi (cond1 dan cond2) yang digabung dengan operator AND. Tabel seperti ini biasa disebut Truth Table atau Tabel Kebenaran.

Cond1	Cond2	Cond1 AND Cond2
TRUE	TRUE	TRUE
TRUE	FALSE	FALSE
FALSE	TRUE	FALSE

Hanya jika kondisi 1 dan kondisi 2 bernilai TRUE maka hasilnya akan TRUE

FALSE	FALSE	FALSE
-------	-------	-------

Contoh:

Bila nilai A=6, B = 3, C= 7 dan D= 4 maka:

If (A > B && C > D) true true	Bernilai TRUE
If (A > B && B > D) true false	Bernilai FALSE
If (A > C && B > D) false false	Bernilai FALSE
If (A > B && B > D) true false	Bernilai FALSE

- c. Operator OR Operator OR menggabungkan dua buah kondisi menjadi satu nilai sedemikian rupa akan bernilai TRUE dengan cukup apabila salah satu kondisi bernilai TRUE. Dan akan bernilai FALSE hanya jika kedua kondisi bernilai FALSE.

Pemakaian dalam program C/C++ atau Java

If (cond1 || cond2)

Atau

If (cond1) || (cond2)

Contoh penulisan dalam bahasa C/C++ atau Java:

```
3) If ( Status == 1 || Umur >=17)
4) If ( Nilai1 >= 60 || Nilai2 >= 70)
```

Syarat sebagai
pemilih: harus
sudah menikah
(status=1) atau
umur tidak
kurang dari 17
tahun(17 tahun
keatas)

Truth Table

Cond1	Cond2	Cond1 OR Cond2
TRUE	TRUE	TRUE
TRUE	FALSE	TRUE
FALSE	TRUE	TRUE
FALSE	FALSE	FALSE

Contoh:

Bila nilai A=6, B = 3, C= 7 dan D= 4 maka:

If (A > B C > D) true true	Bernilai TRUE
If (A > B B > D) true false	Bernilai TRUE
If (A > C B > D) false false	Bernilai FALSE
If (A > B B > D) true false	Bernilai TRUE

1.3 Konversi Multi Condition menjadi Nested If

Berikut ini contoh cara mengkonversi statement if multi condition kedalam bentuk nested if.

Contoh:

```

If ( C1 || C2)
{ S1}
else
{S2}

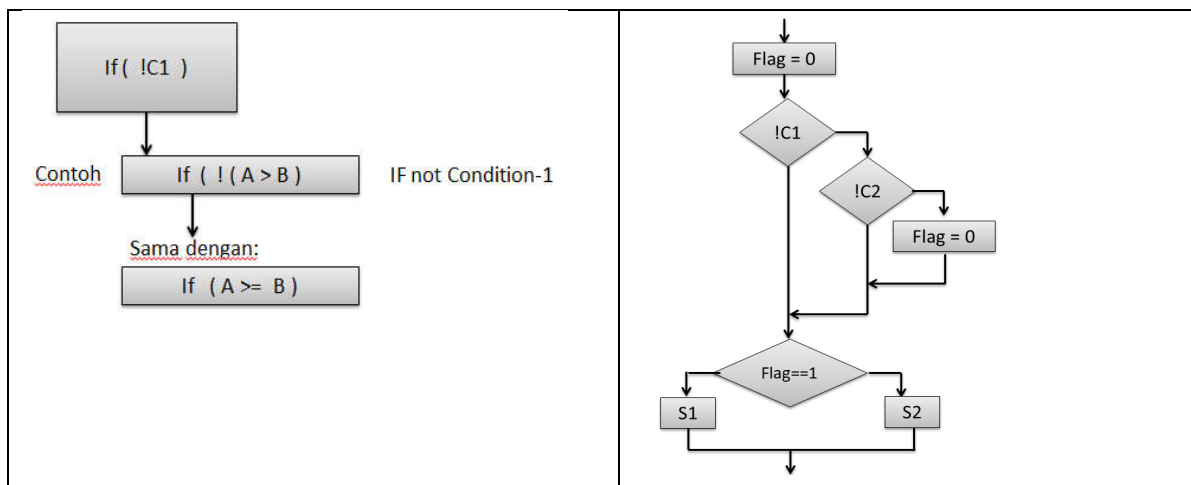
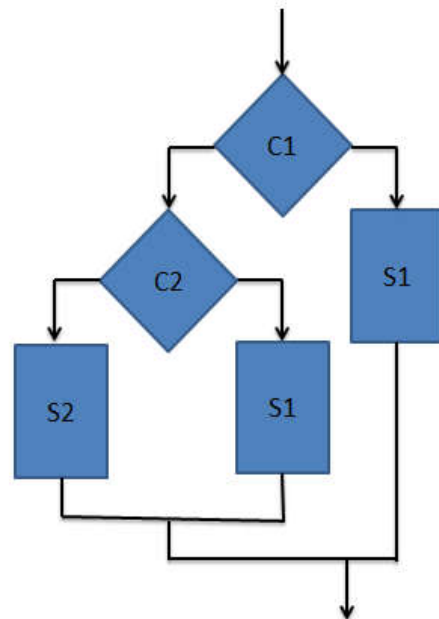
```



```

If (C1)
{S1}
else
{ if (C2)
{S1}
else
{S2}
}

```



1.4 Contoh Program Sederhana Menggunakan Nested If Dan Multi Condition

Berikut contoh soal dengan tiga kondisi:

Susunlah suatu program menggunakan bahasa pemrograman yaitu bahasa C++ untuk menginput tiga bilangan bulat (integer). Dimana ketiga buah bilangan tersebut bernilai tidak sama kemudian mencetak salah satu bilangan yang dinyatakan terbesar!

```

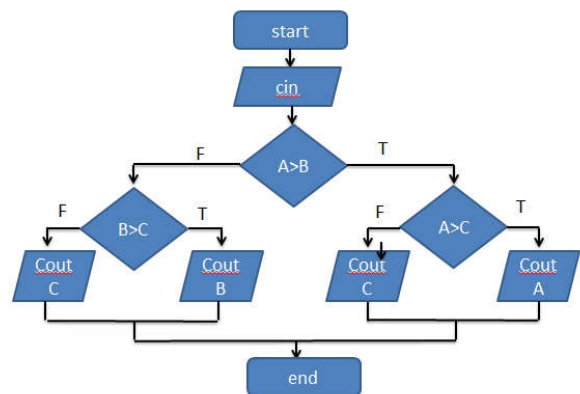
#include <iostream>
using namespace std;
int main()
{
    int pertama, kedua, ketiga;
    cout<<"Input bilangan pertama : ";
    cin>>pertama;
    cout<<"input bilangan kedua : ";
    cin>>kedua;
    cout<<"input bilangan ketiga : ";
    cin>>ketiga;
    if (pertama >kedua)
    {

```

```

C:\Users\user\Documents\Project2.exe
Input bilangan pertama : 11
input bilangan kedua : 12
input bilangan ketiga : 7
12
-----
Process exited after 8.892 seconds with return value 0
Press any key to continue . . .

```

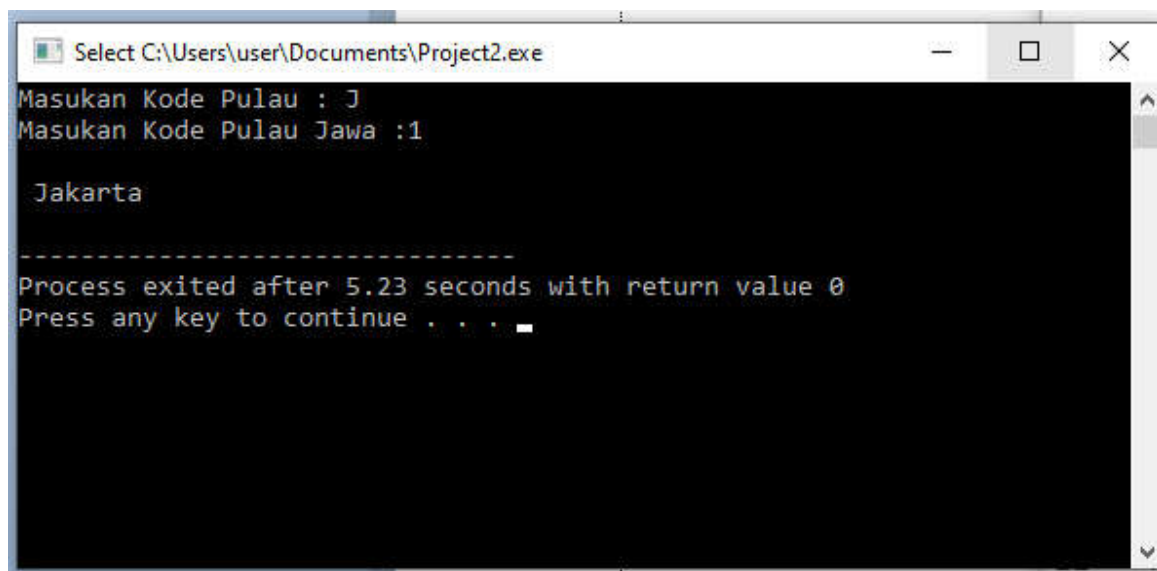


1.5 Switch & Case Berjenjang

Contoh:

```

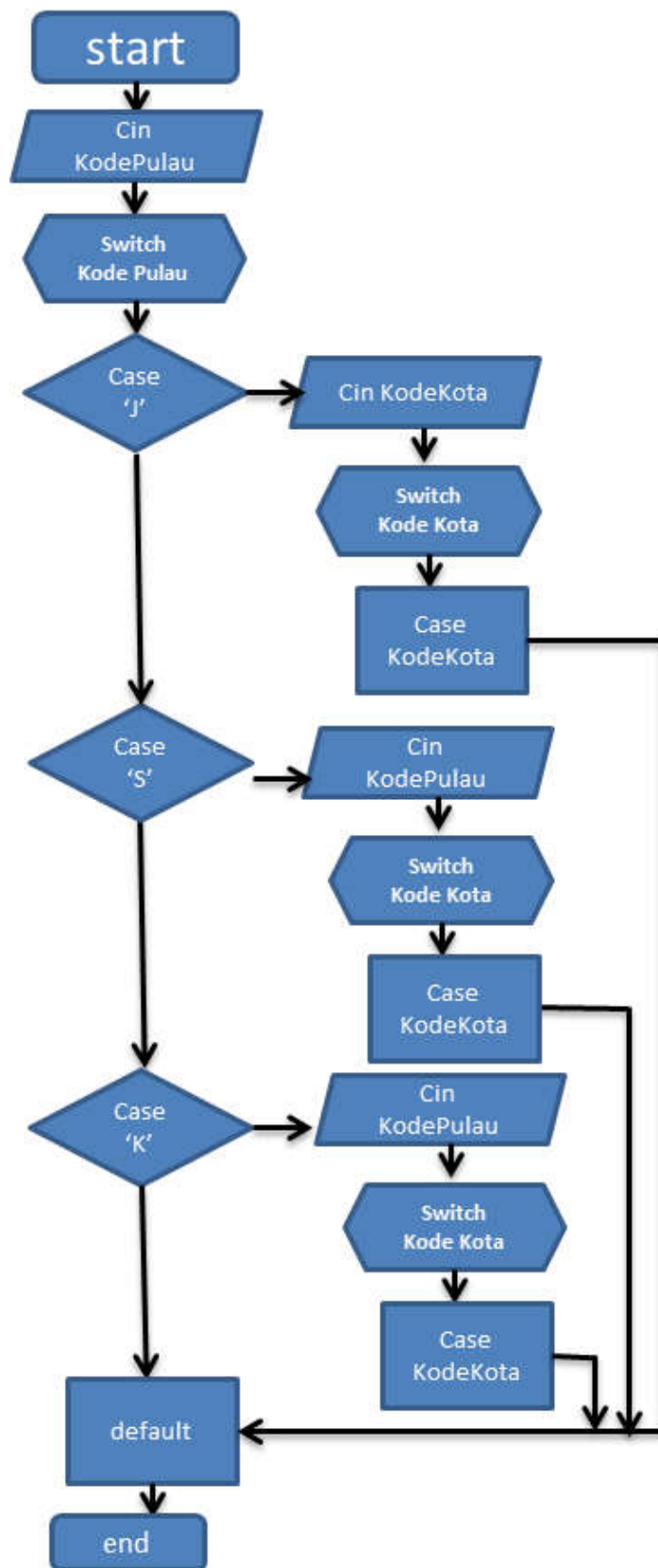
#include <iostream>
using namespace std;
main()
{
    char KodePulau; int KodeKota;
    // char adalah variabel yang isinya karakter diikuti dengan nama variabel
    yang akan dibuat
    cout<<"Masukan Kode Pulau : ";
    //sourir diatas berfungsi untuk menampilkan " Masukan Kode Pulau"
    cin>>KodePulau;
    // berfungsi untu inputan kode pulau sesuai dengan case masing-masing
    switch(KodePulau)
    {
        case 'J':
            cout<<"Masukan Kode Pulau Jawa : ";
            cin>>KodeKota;
            switch (KodeKota)
  
```



```
Select C:\Users\user\Documents\Project2.exe
Masukan Kode Pulau : J
Masukan Kode Pulau Jawa :1

Jakarta

-----
Process exited after 5.23 seconds with return value 0
Press any key to continue . . .
```



Soal Latihan

1. Susunlah program untuk menginput empat buah bilangan bulat kemudian mencetak salah satu bilangan yang nilainya terbesar, serta buat flowchatnya!

Referensi

Charibaldi, N. (2004). *Modul Kuliah Algoritma Pemrograman II Edisi Kedua*. Yogyakarta.

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

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.