



TIB26 – ALGORITMA PEMROGRAMAN



Seleksi Bersarang dan Switch Case

Pertemuan 21, 22

Sub-CPMK

- Mahasiswa mampu menggunakan operasi Seleksi bersarang pada algoritma (C3, A3)

Materi

1. IF Bersarang pada THEN
2. IF Bersarang pada ELSE
3. IF Bersarang pada THEN dan ELSE
4. Ekspresi IF Berurutan
5. Pernyataan Switch-Case

Perhatian

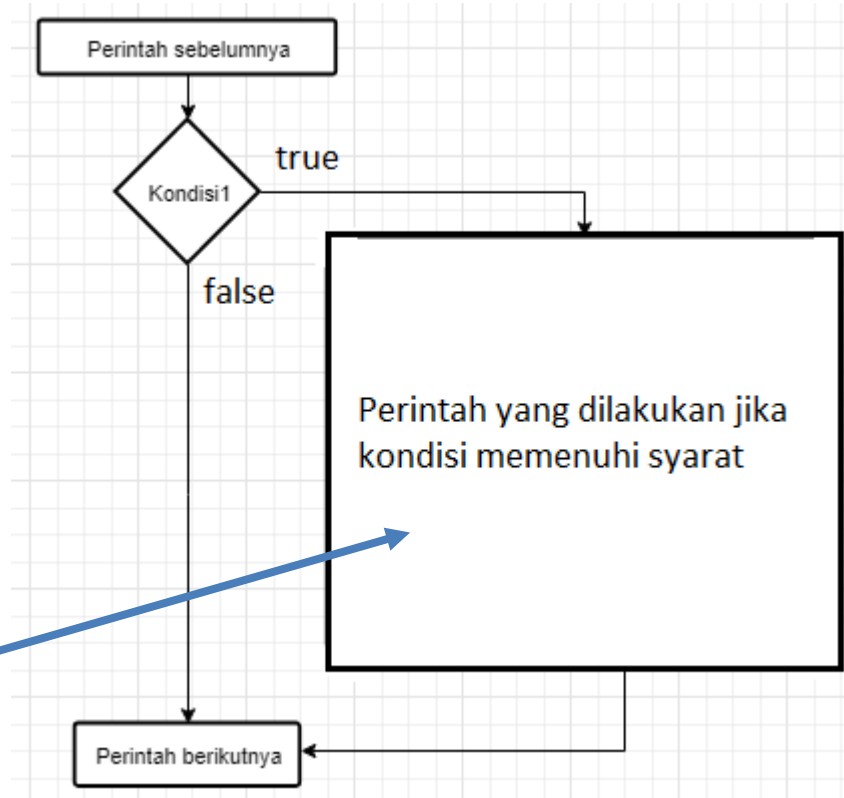
- Tidak disarankan copy-paste kode program dari presentasi ini, karena ada beberapa symbol yang dianggap sebagai Unicode oleh editor yang anda gunakan, sehingga akan dianggap sebagai symbol yang salah oleh compiler, sebaiknya diketik ulang saja



1. IF Bersarang pada THEN

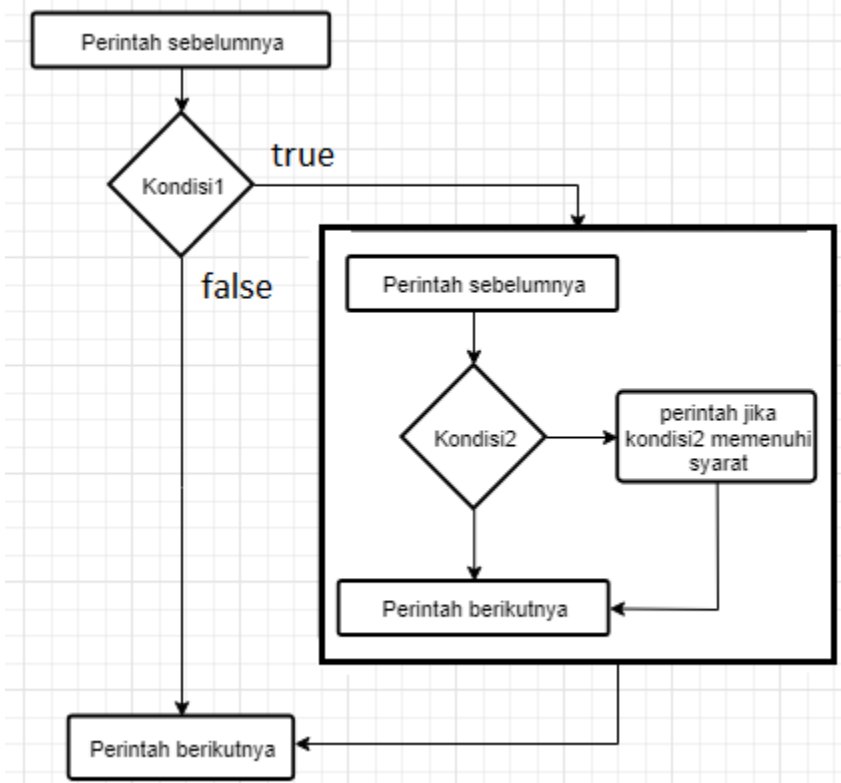
1.1 If bersarang pada then

- Pernyataan if dapat disisipkan sebagai bagian dari if yang lain, menjadi if bersarang
- Andaikan blok sequence perintah dari if kita isi perintah-perintah, maka jika diisi dengan perintah if juga akan menjadi nested if
- Sebagai gambaran anggap saja shape perintah tersebut kita perbesar



1.2 If Bersarang pada then (lanj.)

- Kemudian blok tersebut kita isi dengan pernyataan if yang lain
- Maka terdapat if induk dan if bersarang di dalamnya
- If bersarang dapat ditempatkan pada blok then maupun blok else, tentu saja dengan ekspresi yang berbeda
- If bersarang dapat kita sarangkan sebanyak kebutuhan pemrograman



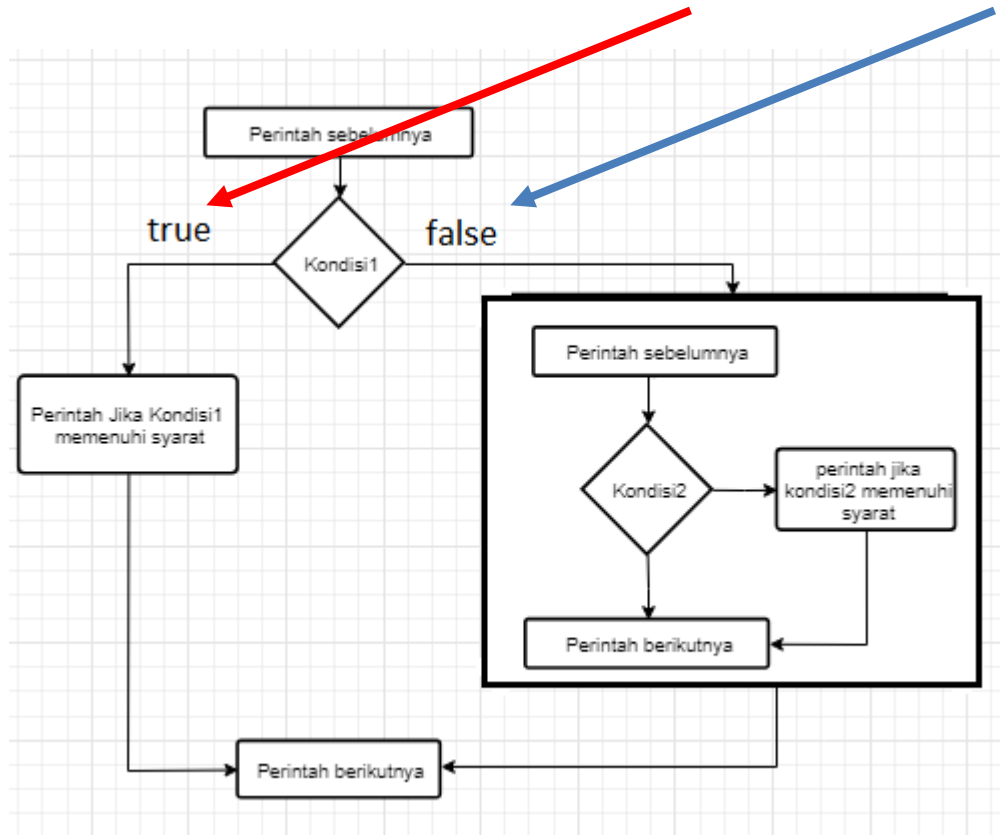
1.3 Penulisan if bersarang pada then

```
if (kondisi1)
{
    if (kondisi2)
    {
        //perintah yang dikerjakan jika kondisi2 memenuhi syarat
    } else
    {
        //perintah yang dikerjakan jika kondisi2 tidak memenuhi syarat
    }
} else
{
    perintah yang dikerjakan jika kondisi1 tidak memenuhi syarat
}
```




2. IF Bersarang pada ELSE

2.1 If bersarang pada else



- Dengan meletakkan if bersarang pada blok 'else', maka kita akan mendapatkan if bersarang dengan ekspresi yang berbeda dengan if bersarang pada blok 'then'
- Pada pola flowchart di samping dapat dilakukan hanya dengan membalikkan nilai true dan false nya saja

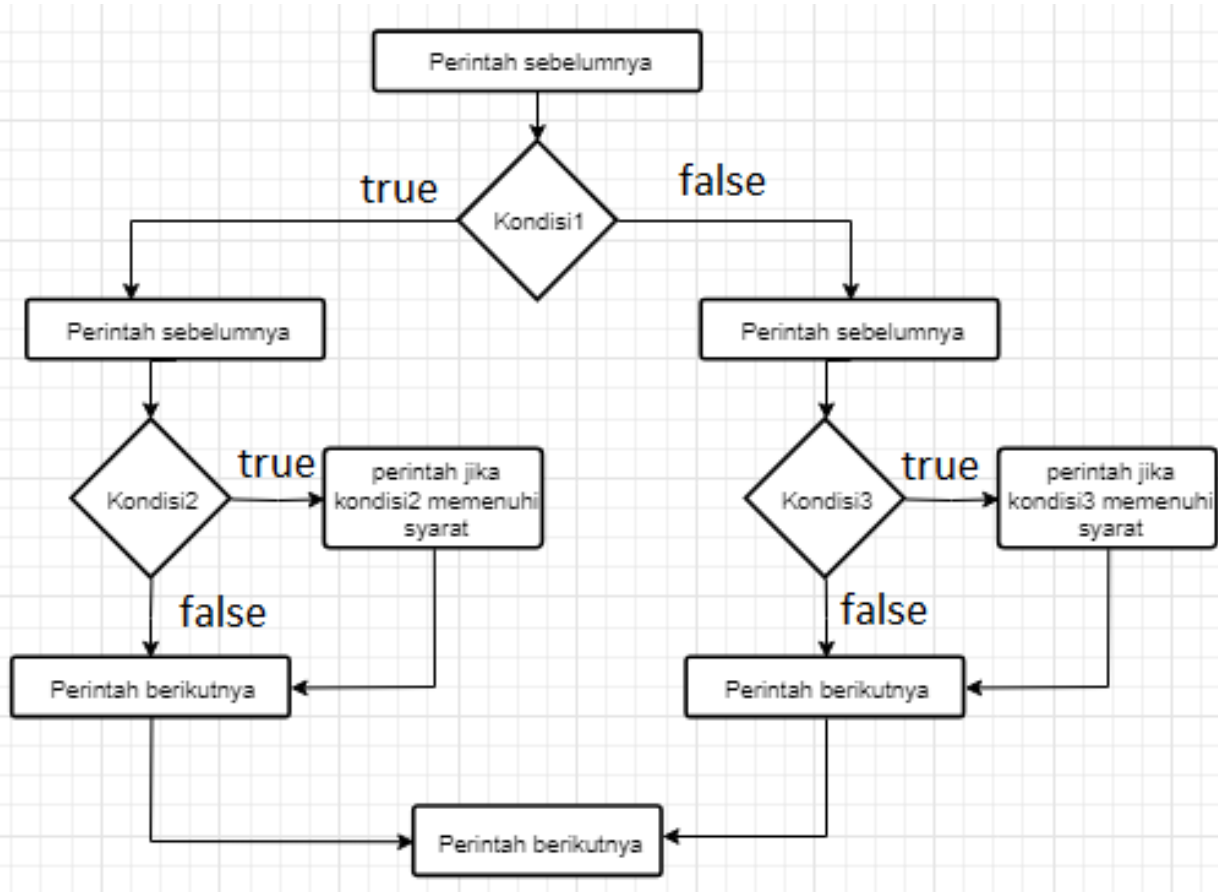
2.2 Penulisan if bersarang pada else

```
if (kondisi1)
{
    perintah yang dikerjakan jika kondisi1 tidak memenuhi syarat
} else
{
    if (kondisi2)
    {
        //perintah yang dikerjakan jika kondisi2 memenuhi syarat
    } else
    {
        //perintah yang dikerjakan jika kondisi2 tidak memenuhi syarat
    }
}
```



3. IF Bersarang pada THEN dan ELSE

3.1 If bersarang pada then dan else



- Ekspresi if bersarang pada then dan else akan mengakibatkan if bersarang memperluas persyaratan pengerjaan percabangan dan memperbanyak percabangan
- Ekspresi ini memerlukan kondisi lebih dari 2, 1 sebagai kondisi pada percabangan utama, sedangkan 2 kondisi untuk kondisi percabangan pada masing-masing blok then dan else

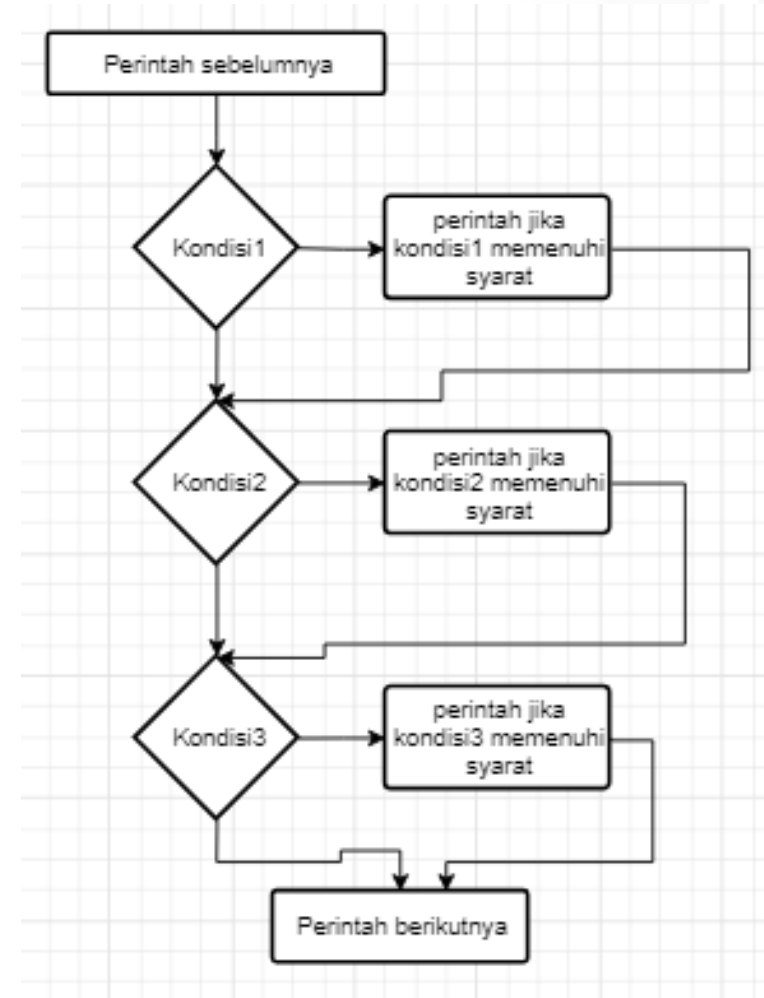
3.2 Penulisan if bersarang pada then dan else

```
if (kondisi1)
{
    if (kondisi2)
    {
        //perintah yang dikerjakan jika kondisi2 memenuhi syarat
    } else
    {
        //perintah yang dikerjakan jika kondisi2 tidak memenuhi syarat
    }
} else
{
    if (kondisi3)
    {
        //perintah yang dikerjakan jika kondisi2 memenuhi syarat
    } else
    {
        //perintah yang dikerjakan jika kondisi2 tidak memenuhi syarat
    }
}
```

4. Ekspresi IF Berurutan

4.1 Ekspresi if berurutan

- Ekspresi sequence dengan persyaratan pada masing-masing proses dapat kita terapkan dengan menempatkan pernyataan percabangan if secara berurutan
- Ekspresi seperti ini nantinya akan sering kita temui pada ekspresi back tracking (akan dibahas pada matakuliah algoritma pemrograman lanjut)



4.2 Ekspresi if berurutan (lanj.)

- Ekspresi tiap percabangan tidak saling berhubungan antara percabangan yang satu dengan percabangan yang lain
- Tetapi ekspresi ini jika didalamnya memproses variable yang sama akan menghasilkan keluaran yang saling mempengaruhi
- Ekspresi seperti ini nantinya akan sering kita temui pada ekspresi back tracking (akan dibahas pada matakuliah algoritma pemrograman lanjut)

4.3 Penulisan if berurutan

```
if(kondisi1)
{
    //perintah jika kondisi1 true
}
if(kondisi2)
{
    //perintah jika kondisi2 true
}
if(kondisi3)
{
    //perintah jika kondisi3 true
}
```



5. Pernyataan Switch-Case

5.1 Pernyataan Switch Case

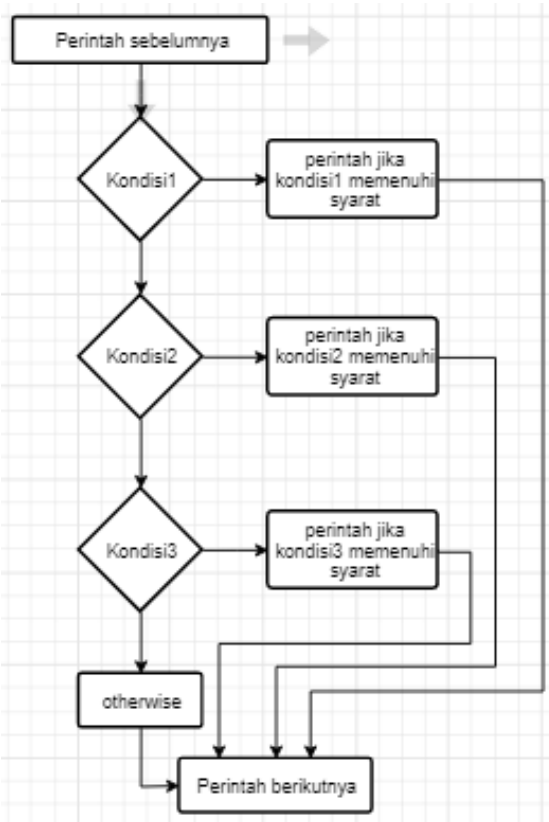
- Switch Case merupakan percabangan yang banyak dengan kondisi lebih dari satu
- Tetapi karena pada teori algoritma percabangan hanya dua yang ditentukan dengan true atau false, maka pada dasarnya switch case merupakan susunan percabangan bersarang pada blok 'else'
- Pada flowchart percabangan hanya memiliki dua keluaran, sehingga pola flowchart disusun dengan diagram percabangan yang disusun berurutan

5.2 Pernyataan Switch Case (lanj.)

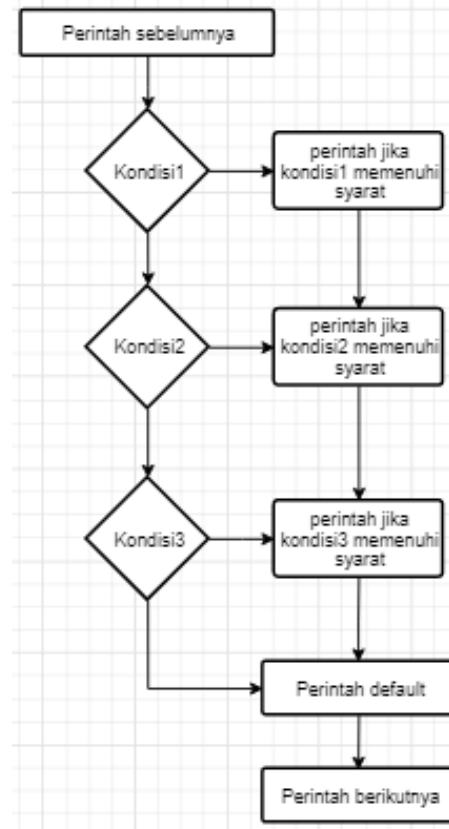
- Perlu diperhatikan, pernyataan switch case pada Bahasa pemrograman berbasis pascal dan berbasis c agak berbeda
- Pada pascal, switch case benar-benar menghasilkan pernyataan percabangan yang terpisah
- Pada c, proses dari case pertama akan dilanjutkan pada case berikutnya, karena itu agar benar-benar menghasilkan switch case yang terpisah, perlu diberi perintah '**break;**' pada tiap-tiap akhir dari case, sehingga proses tidak dilanjutkan ke case berikutnya

5.3 Pola Flowchart Switch Case

Pada Pascal



pada c++



5.4 Switch case c++ tanpa break

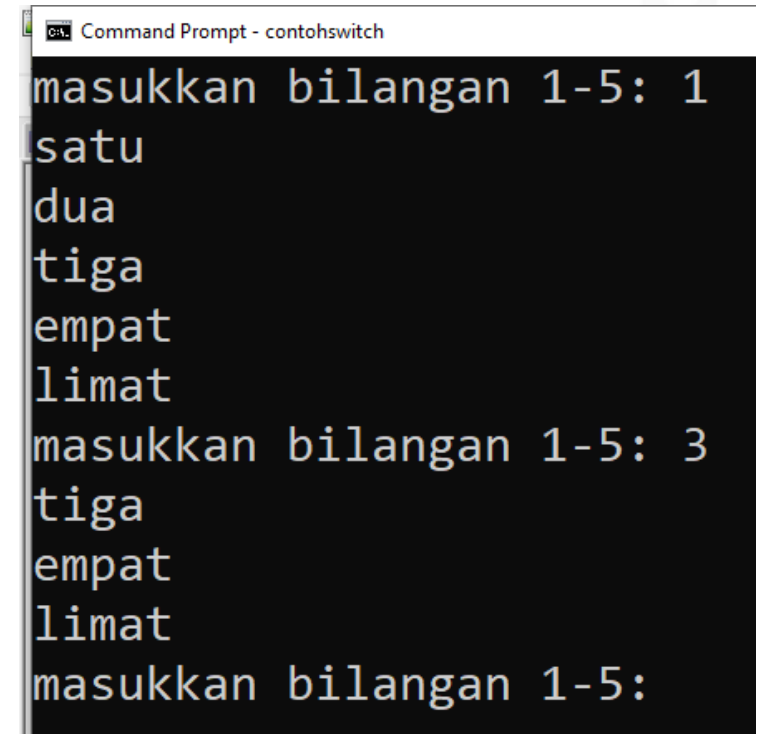
- Perintah break merupakan perintah yang sebaiknya dihindari pada pemrograman terstruktur, tetapi karena perilaku switch-case pada c berbeda dengan pada pascal, maka perintah ini terpaksa dipergunakan, agar proses pada switch-case tidak diteruskan ke case berikutnya sehingga ekspresi switch case pada c dapat seperti switch case pada pascal

5.5 Contoh switch case tanpa break

```
int main(void)
{
    clrscr();
    int jawab=1;
    while((jawab>=1) && (jawab<=5))
    {
        cout<<"masukkan bilangan 1-5: ";
        cin>>jawab;
        switch(jawab)
        {
            case 1:
            {
                cout<<"satu"<<endl;
            }
            case 2:
            {
                cout<<"dua"<<endl;
            }

```

```
            case 3:
            {
                cout<<"tiga"<<endl;
            }
            case 4:
            {
                cout<<"empat"<<endl;
            }
            case 5:
            {
                cout<<"limat"<<endl;
            }
        }
        getch();
    }
}
```



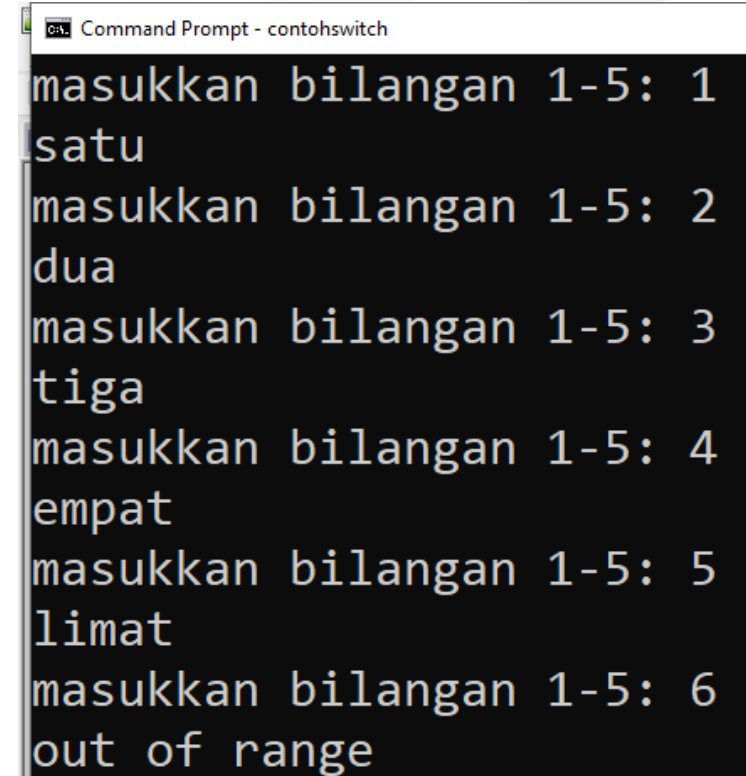
```
Command Prompt - contohswitch
masukkan bilangan 1-5: 1
satu
dua
tiga
empat
limat
masukkan bilangan 1-5: 3
tiga
empat
limat
masukkan bilangan 1-5:
```


5.6 Contoh switch case dengan break

```
int main(void)
{
    clrscr();
    int jawab=1;
    while((jawab>=1) && (jawab<=5))
    {
        cout<<"masukkan bilangan 1-5: ";
        cin>>jawab;
        switch(jawab)
        {
            case 1:
            {
                cout<<"satu"<<endl;
                break;
            }
            case 2:
            {
                cout<<"dua"<<endl;
                break;
            }
        }
    }
}
```

5.6 Contoh switch case dengan break (lanj.)

```
case 3:
{
    cout<<"tiga"<<endl;
    break;
}
case 4:
{
    cout<<"empat"<<endl;
    break;
}
case 5:
{
    cout<<"limat"<<endl;
    break;
}
default:
{
    cout<<"out of range"<<endl;
}
}
getch();
}
```



```
Command Prompt - contohswitch
masukkan bilangan 1-5: 1
satu
masukkan bilangan 1-5: 2
dua
masukkan bilangan 1-5: 3
tiga
masukkan bilangan 1-5: 4
empat
masukkan bilangan 1-5: 5
limat
masukkan bilangan 1-5: 6
out of range
```

Ringkasan

- Pernyataan if dapat disisipkan sebagai bagian dari if yang lain, menjadi if bersarang
- if bersarang dapat ditempatkan pada blok then maupun blok else, dengan ekspresi yang berbeda
- Dengan meletakkan if bersarang pada blok 'else', maka kita akan mendapatkan if bersarang dengan ekspresi yang berbeda dengan if bersarang pada blok 'then'
- Ekspresi if bersarang pada then dan else akan mengakibatkan if bersarang memperluas persyaratan pengerjaan percabangan dan memperbanyak percabangan

Ringkasan (lanj.)

- Ekspresi sequence dengan persyaratan pada masing-masing proses dapat kita terapkan dengan menempatkan pernyataan percabangan if secara berurutan
- Switch Case merupakan percabangan yang banyak dengan kondisi lebih dari satu. Tetapi karena pada teori algoritma percabangan hanya dua yang ditentukan dengan true atau false, maka pada dasarnya switch case merupakan susunan percabangan bersarang pada blok 'else'



Terimakasih

TUHAN Memberkati Anda

Teady Matius Surya Mulyana (tmulyana@bundamulia.ac.id)