



IPB University
— Bogor Indonesia —

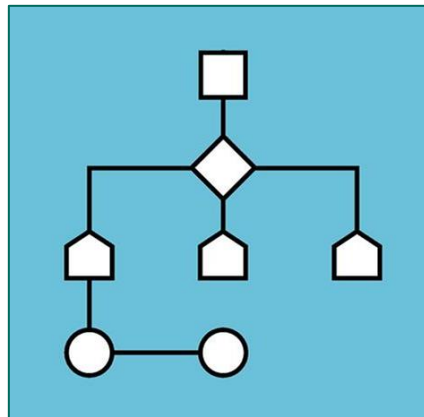
KOM120B #3

Algoritme Kondisional

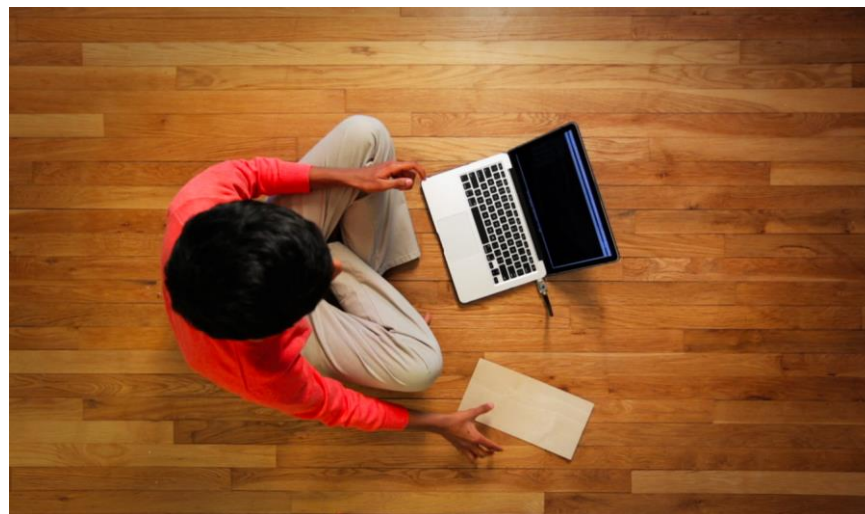
Tim Pengajar

KOM120B – Algoritme dan Dasar Pemrograman

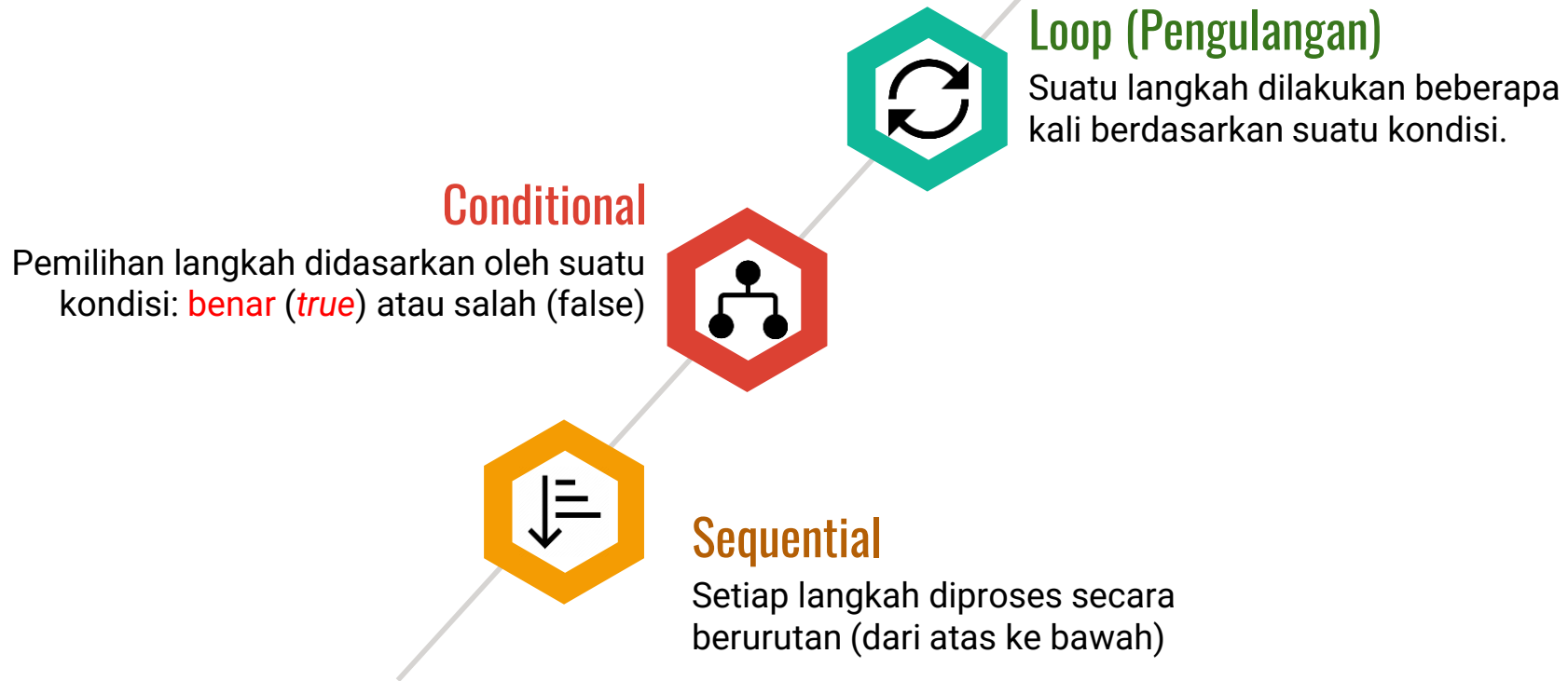
Departemen Ilmu Komputer - FMIPA



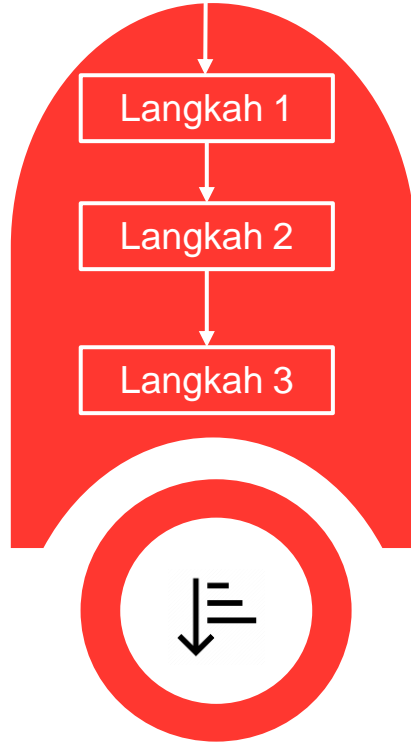
```
7 string sInput;  
8 int iLength, iN;  
9 double dblTemp;  
10 bool again = true;  
21  
22 while (again) {  
23     iN = -1;  
24     again = false;  
25     getline(cin, sInput);  
26     system("cls");  
27     stringstream(sInput) >> dblTemp;  
28     iLength = sInput.length();  
29     if (iLength < 4) {  
30         again = true;  
31         continue;  
32     } else if (sInput[iLength - 3] != '.') {  
33         again = true;  
34         continue;  
35     } while (++iN < iLength) {  
36         if (isdigit(sInput[iN])) {  
37             continue;  
38         } else if (iN == (iLength - 3)) {  
39             continue;  
40         }  
41     }  
42 }
```



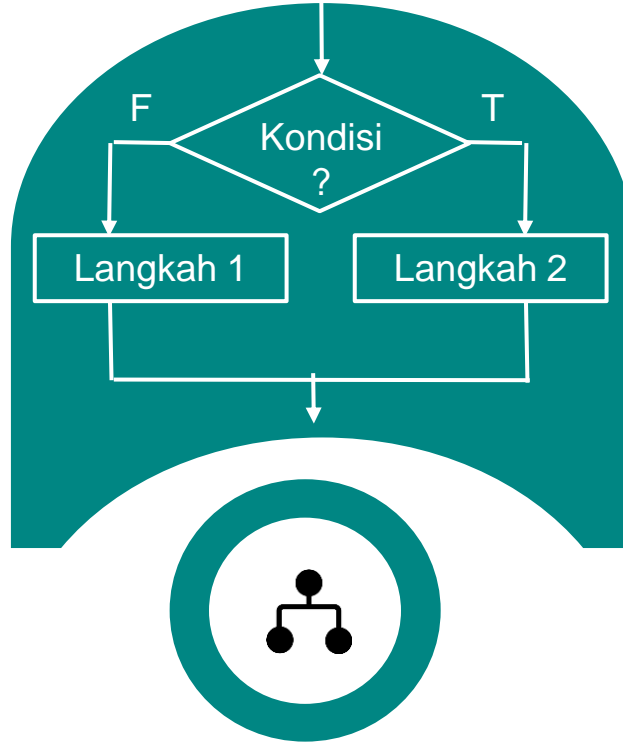
Struktur Algoritme



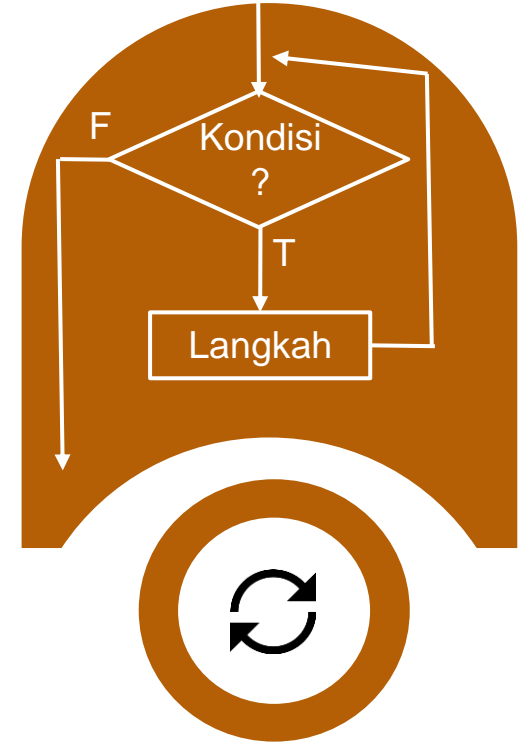
Struktur Algoritme



Sequential



Conditional



Loop

Algoritme Kondisi

Buat algoritme yang memberikan jawaban atas pertanyaan:

Berapa suhu saat ini? Jika suhu kurang dari 20, tuliskan kalimat "Bawa jaket".

- Algoritme:

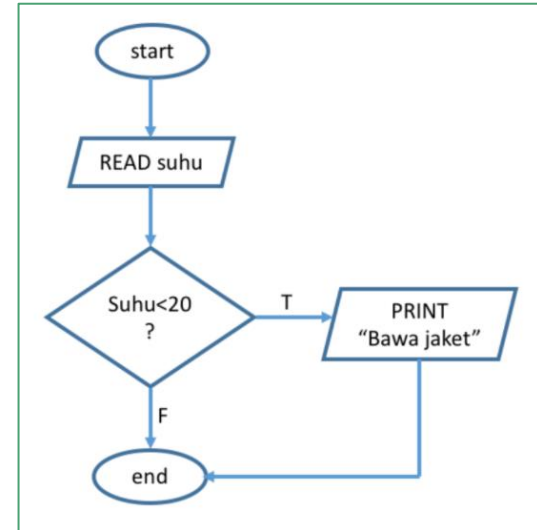
Input : suhu

Jika suhu kurang dari 20, print "Bawa jaket"

- Pseudocode:

read(suhu);

if (suhu<20) print("Bawa jaket");



Algoritme Kondisi

Buat algoritme yang memberikan jawaban atas pertanyaan:

Berapa suhu saat ini? Jika suhu kurang dari 20, tuliskan kalimat "Bawa jaket".

Jika tidak, tuliskan kalimat "Bawa kaos".

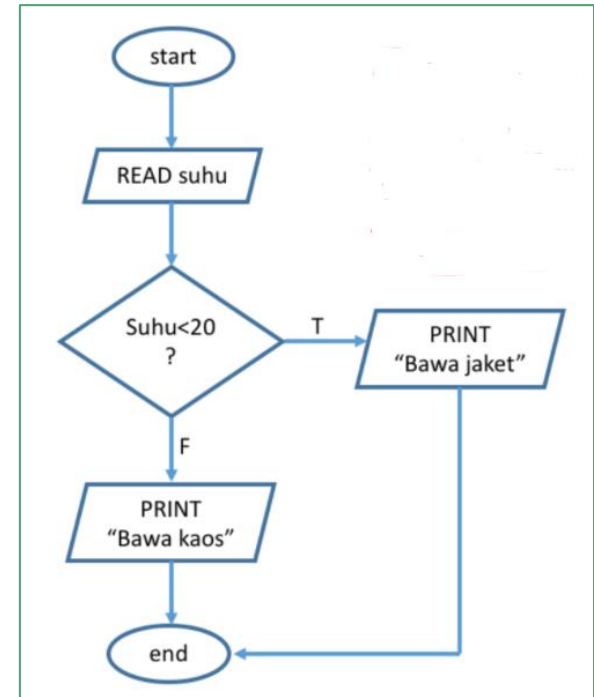
- Algoritme:

Input : suhu

Jika suhu kurang dari 20, print "Bawa jaket"
selainnya, print "Bawa kaos"

- Pseudocode:

```
read(suhu);  
if (suhu<20) print("Bawa jaket");  
else print("Bawa kaos");
```



Dua Bentuk Algoritme Kondisi

- Proses S saat kondisi benar

```
if (kondisi) S;
```

- Proses S1 saat kondisi benar. Selainnya, proses S2.

```
if (kondisi) S1;  
else S2;
```

- Setiap satu **else** harus berpasangan dengan satu **if**.

Ekspresi Logika

Kondisi diimplementasikan sebagai hasil ekspresi logika, yaitu {true, false}

Aljabar	Operator	Contoh Ekspresi	Arti
=	==	(mid == 5)	nilai mid sama dengan 5?
≠	!=	(right != left)	nilai right tidak sama dengan nilai left?
>	>	(kecil > 12.5)	nilai kecil lebih besar dari 12.5?
<	<	(kecil < 12.5)	nilai kecil lebih kecil dari 12.5?
≥	>=	(a >= 10)	nilai a lebih besar atau sama dengan 10?
≤	<=	(a <= 10)	nilai a lebih kecil atau sama dengan 10?
∧	&&	(b>5 && b<10)	jika b>5 dan b<10 maka
∨		(b<=5 b>=10)	jika b<=5 atau b>=10 maka
~ (not)	!	! (x < 10)	negasi dari x kurang dari 10 → berarti x>=5

Ekspresi Logika vs Ekspresi Penugasan

- Ekspresi penugasan

`<variabel> = <ekspresi>;`

- Contoh:

`a=5;`

`b=c=3.15;`

- Apa output potongan program berikut jika input data: 0?

```
scanf ("%d", &a);
```

```
if (a=0) printf("nol\n");
```

```
printf("bukan nol\n");
```


Compound Statement

- Compound statement (pernyataan/instruksi majemuk) adalah satu atau lebih instruksi yang dikelompokkan menjadi satu.
- Algoritme:

jika a kurang dari 5, maka lakukan:

- tambahkan a dengan 10 dan simpan menjadi nilai a baru
- cetak nilai a

selainnya, hitung $c = a - 4$

```
if (a<5)
{
    a=a+10;
    print (a) ;
}
else c=a-4;
```

compound statement

Contoh #1

Buat program untuk memeriksa apakah bilangan masukan merupakan bilangan genap atau ganjil. Jika genap, cetak angka 1. Jika ganjil, cetak angka 0.

- Algoritme:

Input : nilai

jika ($\text{nilai} \% 2 == 0$) print(1)

selainnya print(0).

- Potongan program:

```
int nilai;
```

```
if (nilai%2==0) printf("1\n");
```

```
else printf("0\n");
```

```
return 0;
```

Contoh #2

Buat program untuk menentukan bilangan terbesar dari 3 bilangan bulat masukan.

- Algoritme (1):
Input : a,b,c
jika (a>b && a>c) print(a)
jika (b>a && b>c) print(b)
jika (c>a && c>b) print(c)
- Benarkah algoritme tersebut? Ujilah dengan data masukan:
4 9 2
10 10 5

Contoh #2 : Revisi Algoritme (1)

Buat program untuk menentukan bilangan terbesar dari 3 bilangan bulat masukan.

- Algoritme (2):
Input : a,b,c
jika (a>=b && a>=c) print(a)
jika (b>=a && b>=c) print(b)
jika (c>=a && c>=b) print(c)
- Benarkah algoritme tersebut? Ujilah dengan data masukan:
4 9 2
10 10 5

Contoh #2 : Revisi Algoritme (2)

Buat program untuk menentukan bilangan terbesar dari 3 bilangan bulat masukan.

- Algoritme (3):

Input : a,b,c

```
jika (a>=b && a>=c) print(a)
selainnya jika (b>=c) print(b)
                selainnya print(c)
```

Multiple IF

- Benarkah algoritme tersebut? Ujilah dengan data masukan:

4 9 2

10 10 5

Contoh #2 : Multiple IF

Buat program untuk menentukan bilangan terbesar dari 3 bilangan bulat masukan.

```
#include <stdio.h>
int main()
{
    int a,b,c;
    scanf("%d %d %d", &a, &b, &c);
    if (a>=b && a>=c) printf("%d\n", a);
    else if (b>=c)    printf("%d\n", b);
        else printf("%d\n", c);
    return 0;
}
```

Nested IF

- IF bersarang → IF di dalam IF
- Contoh potongan program:

```
scanf("%d %d", &a, &b);  
if (a>b)  
{  
    c=a-b;  
    if (c<5) printf("%d\n", a);  
    else printf("%d\n", b);  
}  
else printf("%d\n", b-a);
```

Latihan #1 : Output Program

Apa output potongan program berikut jika diberikan input data:

15 10 5

```
scanf("%d %d %d", &a, &b, &c);  
if (a>b>c) printf("menurun\n");  
else printf("tidak menurun\n");
```

Bagaimana jika diberikan input data:

3 2 1

Latihan #2 : Status Kelulusan

Buat program untuk menentukan status kelulusan dari suatu mata kuliah yang ditentukan berdasarkan nilai UTS dan UAS (keduanya bilangan bulat 0 sampai dengan 100) dengan ketentuan sebagai berikut:

- Jika UTS dan UAS keduanya lebih besar dari 40 dan rata-ratanya lebih besar dari 60, maka dinyatakan LULUS.
- Jika UTS atau UAS ada yang bernilai lebih kecil atau sama dengan 40, sedangkan rata-ratanya lebih besar dari 60, dinyatakan LULUS BERSYARAT.
- Jika rata-ratanya kurang dari 60, dinyatakan TIDAK LULUS

Latihan #3 : Tebak Bilangan

Buat program membaca dua bilangan bulat a dan b, kemudian membaca sebuah bilangan bulat c dan menentukan dimana posisi bilangan c berada dalam sebuah garis bilangan. Jika bilangan c berada di sebelah kiri a, maka program menuliskan output **kiri**, jika di tengah antara a dan b maka cetak **tengah**, jika di kanan b maka cetak **kanan**. Sedangkan jika tepat di posisi a atau b maka cetak teks **tepi**.

Contoh Masukan

```
3 10
6
```

Contoh Keluaran

```
tengah
```