

04

Introduction to Algorithm

LOOP

sebagai pembentuk

kerangka dasar

Algoritma

apakah

LOOP

itu



LOOP adalah sekelompok **instruksi** yang dikerjakan secara berulang-ulang

LOOP adalah suatu **proses** yang terjadi secara berulang-ulang

control statement

for dan while

sebagai pembentuk

LOOP

contoh

Penggunaan

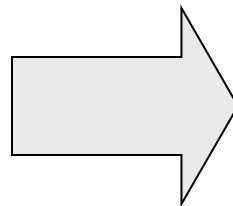
for() dan while()

Contoh-1 :

```
#include<iostream.h>
void main()
{int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\nJakarta";
  }
  cout << "\nSelesai";
}
```

```
#include<iostream.h>
void main()
{int I;
  I=1;
  while( I<=5 )
  {
    cout << "\nJakarta";
    I = I + 1;
  }
  cout << "\nSelesai";
}
```

Bila kedua program diatas dijalankan, (diRUN) maka keduanya akan tecetak:



Jakarta
Jakarta
Jakarta
Jakarta
Jakarta
Selesai

Bentuk Umum

```
for ( init; cond; chng of cond )
{
  -
  -
  - loop
  -
  -
}
```

Loop adalah sekumpulan instruksi yang rencananya akan dikerjakan secara berulang-ulang

cond = condition

Suatu pernyataan yang mengandung nilai **BENAR** (true) atau **SALAH** (False)

init = inisialisai

Instruksi pemberian suatu nilai yang mempengaruhi nilai condition. Pada proses yang normal, pemberian nilai awal ini akan menyebabkan condition bernilai true.

Instruksi ini hanya pernah satu kali dilaksanakan, yaitu hanya pada saat awal

```
init;
while ( cond )
{
  -
  - loop
  -
  -
  chng of cond
}
```

Chng of cond =

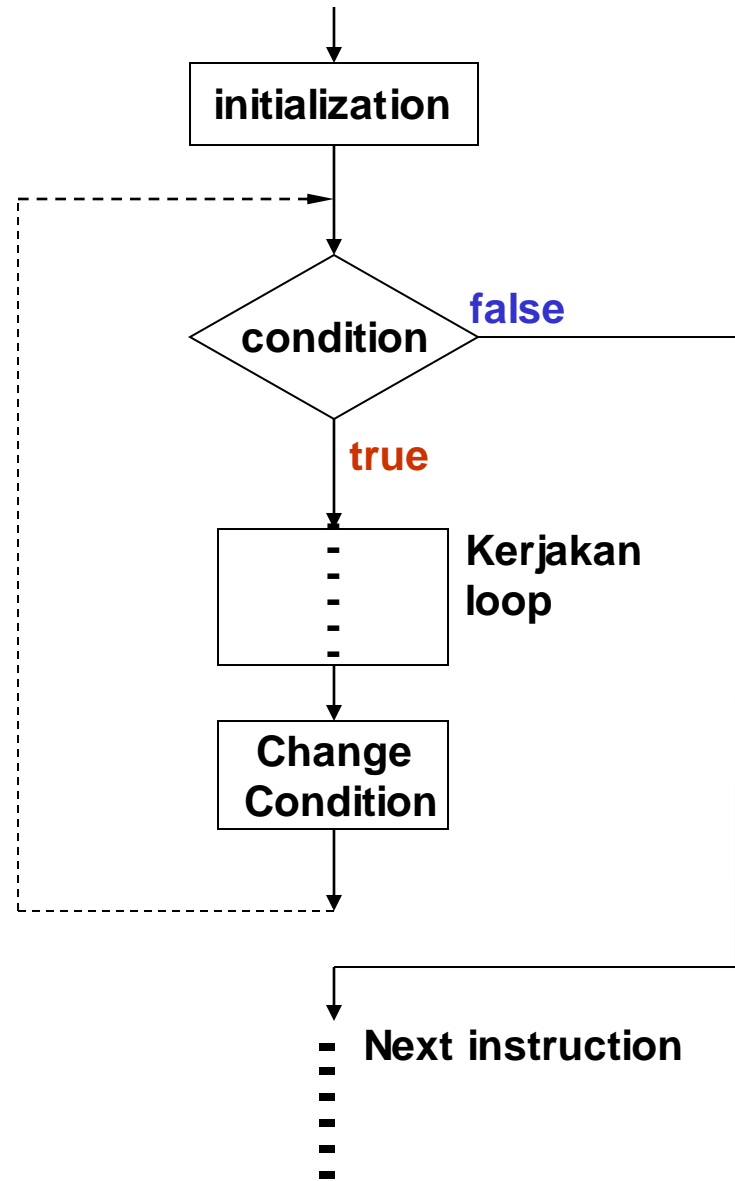
Change of condition

Suatu instruksi yang dapat mempengaruhi nilai condition. Pada proses yang normal, perubahan nilai disini suatu saat akan membuat nilai condition = false

Bentuk Umum

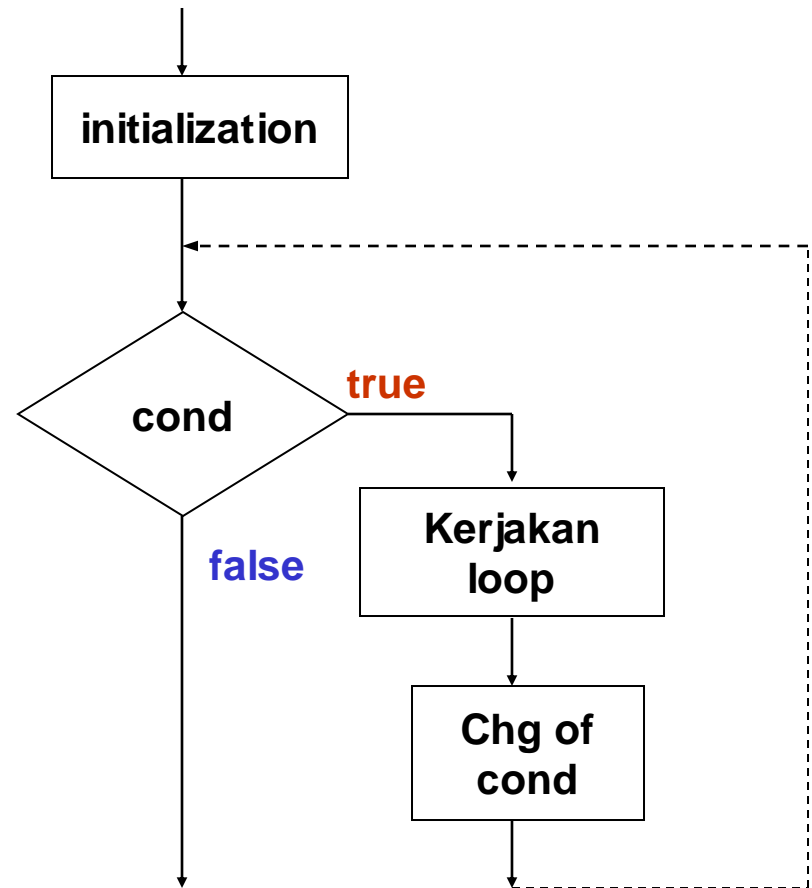
```
for ( init; cond; chng of cond )  
{  
  -  
  - loop  
  -  
}
```

```
init;  
while ( cond )  
{  
  -  
  - loop  
  -  
  chng of cond  
}
```



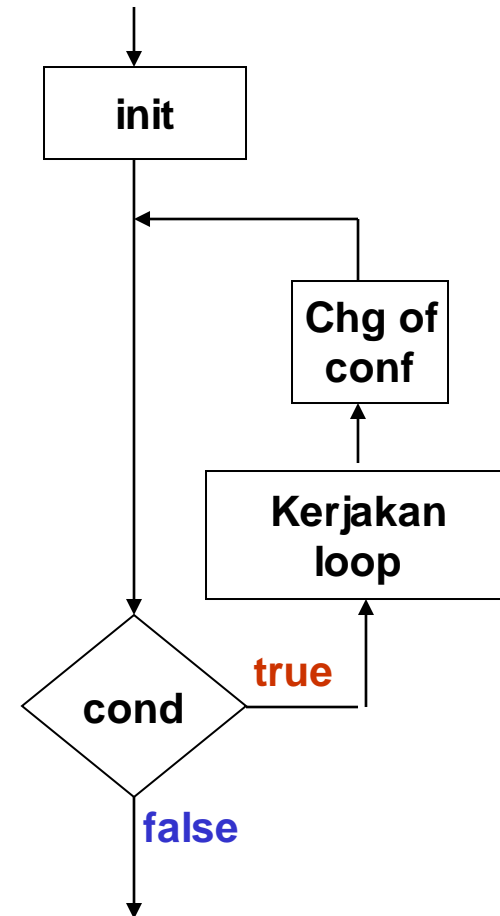

```
for ( init; cond; chng of cond )
{
  - loop
}
```

```
init;
while ( cond )
{
  - loop
  chng of cond
}
```



```
for ( init; cond; chng of cond )  
{  
  -  
  - loop  
  -  
}
```

```
init;  
while ( cond )  
{  
  -  
  - loop  
  -  
  chng of cond  
}
```



for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    loop
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    loop
    I=I+1;
  }
}
```

Berapa Kali Loop Dikerjakan ?

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    loop
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    loop
    I=I+1;
  }
}
```

Jawab : 5 kali

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\nJakarta" ;
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    cout << "\nJakarta";
    I=I+1;
  }
}
```

Apa yang tercetak bila program diatas dikerjakan

Contoh-2 :

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\nJakarta" ;
  }
}
```

Tercetak : Jakarta
Jakarta
Jakarta
Jakarta
Jakarta

```
while()
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    cout << "\nJakarta";
    I=I+1;
  }
}
```

Tercetak : Jakarta
Jakarta
Jakarta
Jakarta
Jakarta

Contoh-3 :

for ()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\n" << I;
  }
}
```

while ()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    cout << "\n" << I;
    I=I+1;
  }
}
```

Apa yang tercetak bila program diatas dijalankan

for ()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\n" << I;
  }
}
```

Tercetak : 1
2
3
4
5

while ()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    cout << "\n" << I;
    I=I+1;
  }
}
```

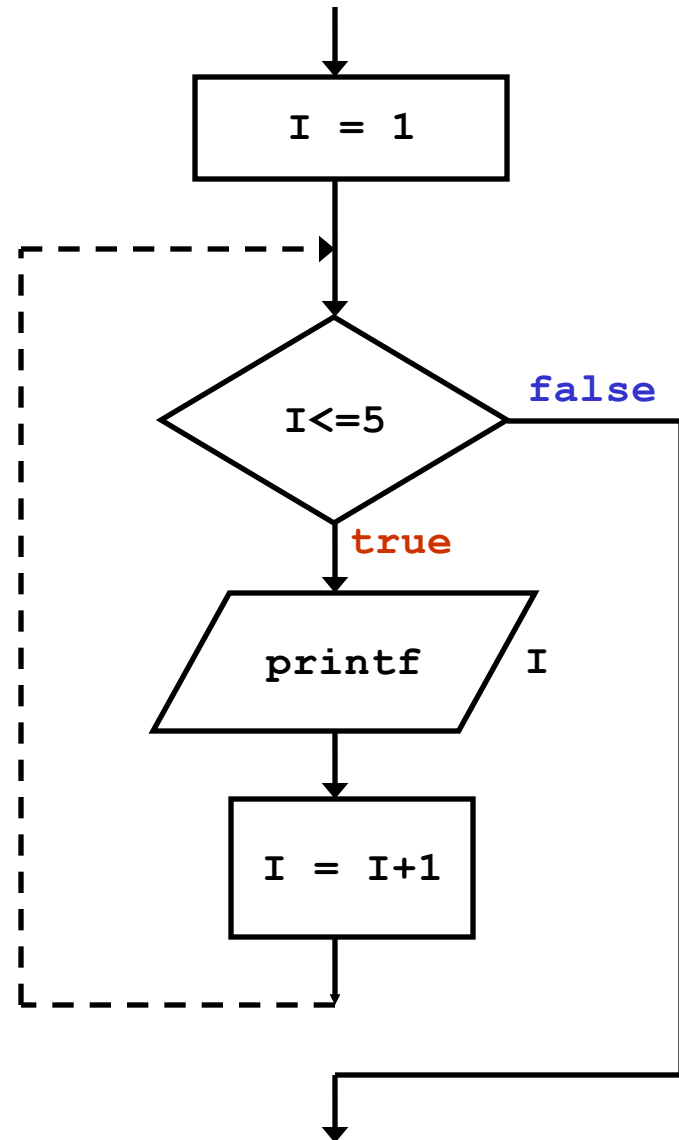
Tercetak : 1
2
3
4
5

I	Tercetak
1	1
2	2
3	3
4	4
5	5
6	

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout << "\n" << I;
  }
}
```

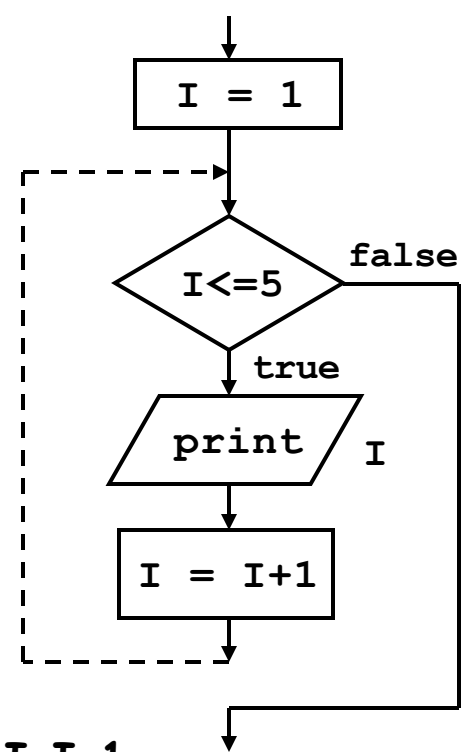
Tercetak : 1
2
3
4
5



for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
    {cout << "\n" << I;
    }
}
```

Tercetak : 1
2
3
4
5



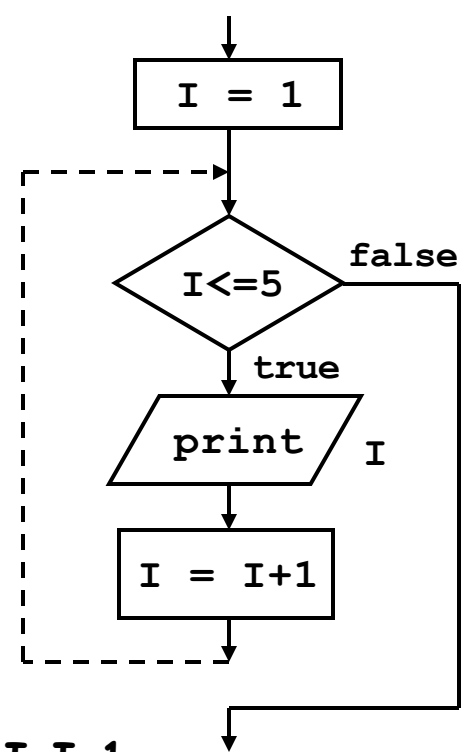
Perkembangan nilai I

nilai I	Kondisi I <= 5	Tercetak oleh cout << I	Oleh I=I+1 nilai I menjadi:
1	True	1	2
2	True	2	3
3	True	3	4
4	True	4	5
5	True	5	6
6	False	Keluar dari loop	

`while()`

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while(I<=5)
  { cout << "\n" << I;
    I=I+1;
  }
}
```

Tercetak : 1
2
3
4
5



Perkembangan nilai I

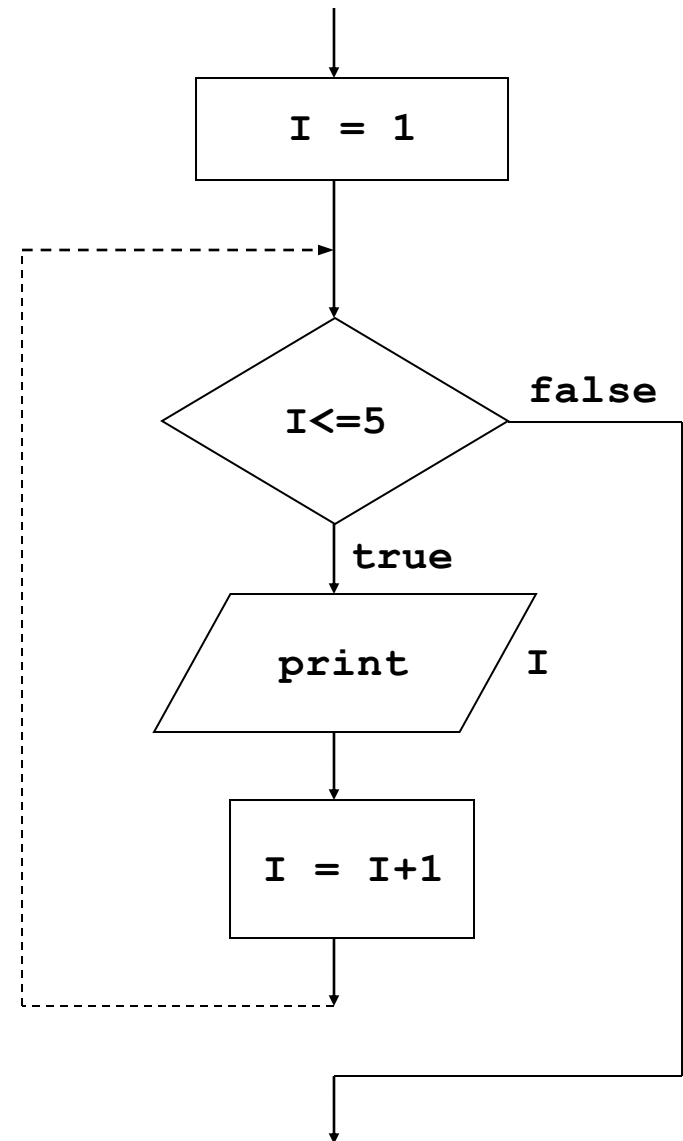
nilai I	Kondisi I <= 5	Tercetak oleh <code>cout << I</code>	Oleh <code>I=I+1</code> nilai I menjadi:
1	True	1	2
2	True	2	3
3	True	3	4
4	True	4	5
5	True	5	6
6	False	Keluar dari loop	

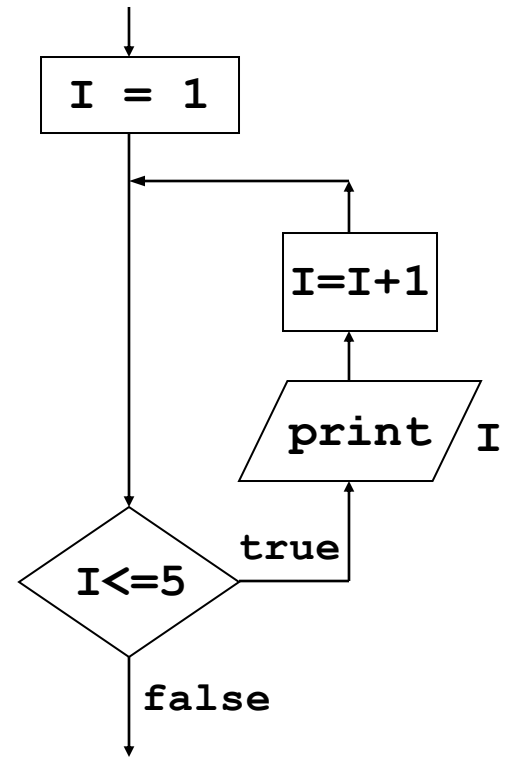
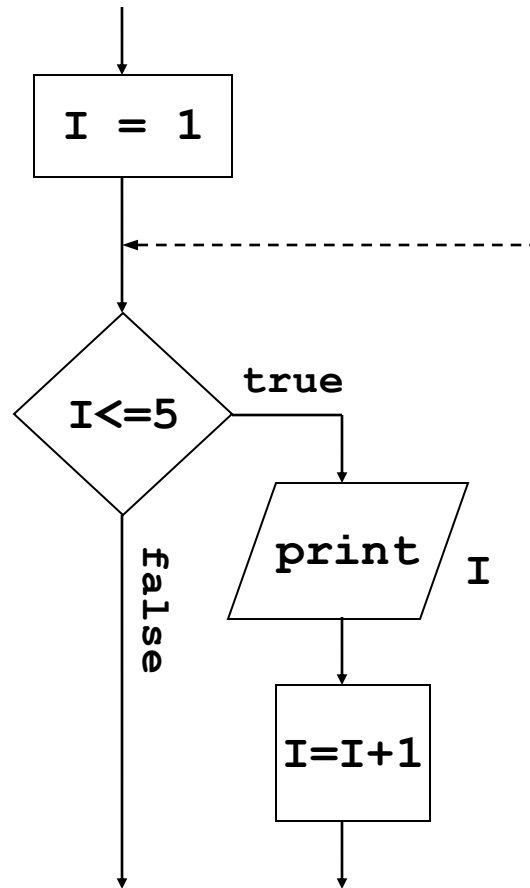
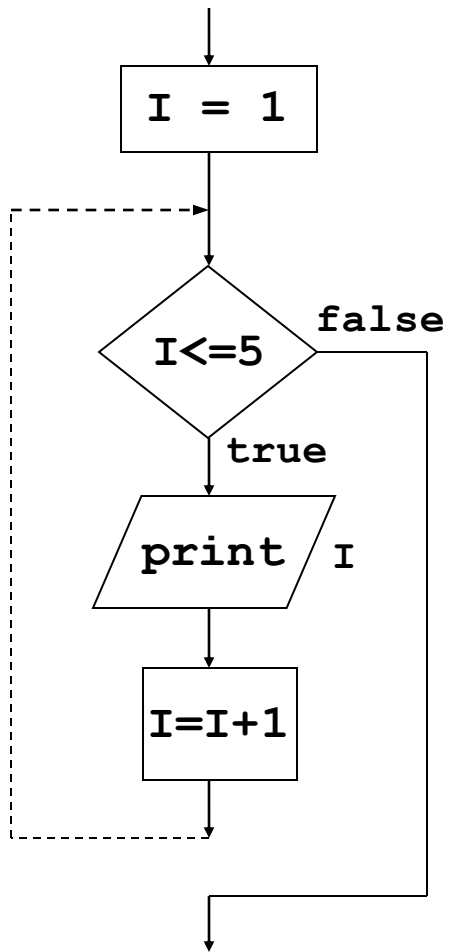
for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
    { cout << "\n" << I;
    }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while(I<=5)
    { cout << "\n" << I;
      I=I+1;
    }
}
```





Perhatikan kembali Contoh-1 sebelumnya
sebagai berikut :

for()

```
#include<iostream.h>
void main()
{ int I;
  for ( I=1 ; I<=5 ; I=I+1 )
  {
    cout << "\n" << I;
  }
}
```

Buku literatur selalu
menggunakan **I++**
untuk **I = I+1**

for()

```
#include<iostream.h>
void main()
{ int I;
  for ( I=1 ; I<=5 ; I++ )
  {
    cout << "\n" << I;
  }
}
```

Beberapa Cara penulisan loop dengan for() untuk Contoh-1 diatas, yang menghasilkan efek yang sama :

```
#include<iostream.h>
void main()
{ int I;
  for ( I=1 ; I<=5 ; I=I+1 )
  {
    cout << "\n" << I;
  }
}
```

```
{ int I;
  for ( I=1 ; I<=5 ; I++ )
    cout << "\n" << I;
}
```

Bila instruksi dalam loop hanya ada 1 instruksi, maka boleh tidak diapit oleh tanda kurung.

```
{ int I;
  for(I=1; I<=5; I++) cout << "\n" << I;
}
```

Loop boleh langsung ditulis pada baris yang sama dengan for().

Perhatikan kembali loop dengan while sebagai berikut:

`while()`

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
    {cout <<"\n"<< I;
      I++;
    }
}
```

dapat ditulis menjadi:

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
    {cout <<"\n"<< I++;
    }
}
```


Contoh-4 :

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout <<"\n " << I*2;
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while(I<=5)
  {
    cout <<"\n " << I*2;
    I=I+1;
  }
}
```

Apa yang tercetak bila program diatas dikerjakan

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout <<"\n " << I*2;
  }
}
```

Tercetak : 2
4
6
8
10

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while(I<=5)
  {
    cout <<"\n " << I*2;
    I=I+1;
  }
}
```

Tercetak : 2
4
6
8
10

for()

```
#include<iostream.h>
void main()
{ int I;
  for(I=1; I<=5; I=I+1)
  {
    cout <<"\n " << I*2;
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I;
  I = 1;
  while (I<=5)
  {
    cout <<"\n " << I*2;
    I=I+1;
  }
}
```

Perkembangan nilai I

nilai I	Kondisi I <= 5	Tercetak oleh Cout << I*2	Oleh I=I+1 nilai I menjadi:
1	True	2	2
2	True	4	3
3	True	6	4
4	True	8	5
5	True	10	6
6	False	Keluar dari loop	

Perhatikan :

15

A

```
cout << A+2;
```

Akan mencetak : 17

Tapi isi variabel A tidak berubah

Contoh-5 :

for()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  for(I=1; I<=5; I=I+1)
  {
    cout << N;
    N = N + 2;
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  I = 1;
  while (I<=5)
  {
    cout << N;
    N = N + 2;
    I=I+1;
  }
}
```

Apa yang tercetak bila program diatas Dilaksanakan

for()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  for(I=1; I<=5; I=I+1)
  {
    cout << N;
    N = N + 2;
  }
}
```

Tercetak : 8
10
12
14
16

while()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  I = 1;
  while (I<=5)
  {
    cout << N;
    N = N + 2;
    I=I+1;
  }
}
```

Tercetak : 8
10
12
14
16

for()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  for(I=1; I<=5; I=I+1)
  {
    cout << N;
    N = N + 2;
  }
}
```

while()

```
#include<iostream.h>
void main()
{ int I, N;
  N = 8;
  I = 1;
  while (I<=5)
  {
    cout << N;
    N = N + 2;
    I=I+1;
  }
}
```

Perkembangan nilai I

N	nilai I	Kondisi I <= 5	Tercetak oleh cout << N	Oleh N=N+2 nilai N menjadi:	Oleh I=I+1 nilai I menjadi:
8	1	True	8	10	2
10	2	True	10	12	3
12	3	True	12	14	4
14	4	True	14	16	5
16	5	True	16	18	6
18	6	False	Keluar dari loop		

Soal

14a

Susun Algoritma yang ditulis dalam Bahasa C++ untuk mencetak 10 suku pertama deret berikut ini :

1, 2, 3, 4, 5,

Bila algoritma benar, maka tercetak :

1
2
3
4
5
6
7
8
9
10

Cara-1

```
#include<iostream.h>
void main()
{ int I;
  for( I=1; I<=10; I=I+1 )
    { cout << "\n" << I ;
    }
}
```

I	Tercetak :
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	

I terakhir nilainya = 11,
Tapi tidak ikut dicetak
Karena sudah keluar dari loop

Cara-2

```
#include<iostream.h>
void main()
{ int I, N;
  for( I=1; I<=10; I=I+1 )
  { N = I;
    cout << "\n" << N ;
  }
}
```

I	N
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	10
11	

Cara-3

```
#include<iostream.h>
void main()
{ int I, N;
  N = 1;
  for( I=1; I<=10; I=I+1 )
  { cout << "\n" << N ;
    N = N + 1;
  }
}
```

I	N
1	1
2	2
3	3
4	4
5	5
6	6
7	7
8	8
9	9
10	12
11	11

N awalnya =1
dan selalu ditambah 1

Nilai N pernah = 11
tapi tidak ikut dicetak
karena sudah keluar dari loop

Perkembangan nilai I

N	nilai I	Kondisi I <= 10	Tercetak oleh Cout << N	Oleh N=N+1 nilai N menjadi:	Oleh I=I+1 nilai I menjadi:
1	1	True	1	2	2
2	2	True	2	3	3
3	3	True	3	4	4
4	4	True	4	5	5
5	5	True	5	6	6
6	6	True	6	7	7
7	7	True	7	8	8
8	8	True	8	9	9
9	9	True	9	10	10
10	10	True	10	11	11
11	11	False	Keluar dari loop		

Soal

14b

Susun Algoritma yang ditulis dalam Bahasa C++ untuk mencetak 10 suku pertama deret berikut ini :

1, 3, 5, 7, 9,

Bila algoritma benar, maka tercetak :

1
3
5
7
9
11
13
15
17
19

Cara-1

```
#include<iostream.h>
void main()
{ int I;
  for( I=1; I<=19; I=I+2 )
    { cout << "\n" << I ;
    }
}
```

I	Tercetak :
1	1
3	3
5	5
7	7
9	9
11	11
13	13
15	15
17	17
19	19
21	

I terakhir nilainya = 21,
Tapi tidak ikut dicetak
Karena sudah keluar dari loop

Cara-2

```
#include<iostream.h>
void main()
{ int I, N;
  for( I=1; I<=10; I=I+1 )
  { N = I * 2 - 1;
    cout << "\n" << N;
  }
}
```

I	N
1	1
2	3
3	5
4	7
5	9
6	11
7	13
8	15
9	17
10	19
11	

Cara-3

```
#include<iostream.h>
void main()
{ int I, N;
  N = 1;
  for( I=1; I<=10; I=I+1 )
  { cout << "\n" << N;
    N = N + 2;
  }
}
```

I	N
1	1
2	3
3	5
4	7
5	9
6	11
7	13
8	15
9	17
10	19
11	21

N awalnya =1
dan selalu ditambah 2

Nilai N pernah = 21
tapi tidak ikut dicetak
karena sudah keluar dari loop

Soal

14c

Susun Algoritma yang ditulis dalam Bahasa C++ untuk mencetak 10 suku pertama deret berikut ini :

1, 2, 4, 8, 16,

Bila algoritma benar, maka tercetak :

1
2
4
8
16
32
64
128
256
512

Cara-1

```
#include<iostream.h>
void main()
{ int I;
  for( I=1; I<=512; I=I*2 )
    { cout << "\n" << I ;
    }
}
```

I	Tercetak :
1	1
2	2
4	4
8	8
16	16
32	32
64	64
128	128
256	256
512	512
1024	

Cara-2

```
#include<iostream.h>
#include<math.h>
void main()
{ int I, N;
  for( I=1; I<=10; I=I+1 )
    { N = pow(2, (I-1));
      cout << "\n" << N ;
    }
}
```

pow(2, (I-1))

Maksudnya **2** pangkat **(I-1)**

Agar dapat menggunakan fungsi matematik **pow** (power), harus menambah : **# include<math.h>**

I	N
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128
9	256
10	512
11	

Cara-3

```
#include<iostream.h>
void main()
{ int I, N;
  N = 1;
  for( I=1; I<=10; I=I+1 )
    { cout << "\n" << N ;
      N = N * 2;
    }
}
```

N awalnya =1
dan selalu dikali 2

I	N
1	1
2	2
3	4
4	8
5	16
6	32
7	64
8	128
9	256
10	512
11	1024

Nilai **N** pernah = **1024**
tapi tidak ikut dicetak
karena sudah keluar dari loop

Soal

14d

Susun Algoritma yang ditulis dalam Bahasa C++ untuk mencetak 10 suku pertama deret berikut ini :

5, 8, 11, 14, 17,

Bila algoritma benar, maka tercetak :

5
8
11
14
17
20
23
26
29
32

Cara-1

```
#include<iostream.h>
void main()
{ int I;
  for( I=5; I<=32; I=I+3 )
    { cout << "\n" << I ;
    }
}
```

I	Tercetak :
5	5
8	8
11	11
14	14
17	17
20	20
23	23
26	26
29	29
32	32
35	

Cara-2

```
#include<iostream.h>
void main()
{ int I, N;
  for( I=1; I<=10; I=I+1 )
  { N = I*3 + 2 ;
    cout << "\n" << N ;
  }
}
```

I	N
1	5
2	8
3	11
4	14
5	17
6	20
7	23
8	26
9	29
10	32
11	

Cara-3

```
#include<iostream.h>
void main()
{ int I, N;
  N = 5;
  for( I=1; I<=10; I=I+1 )
  { cout << "\n" << N ;
    N = N + 3;
  }
}
```

I	N
1	5
2	8
3	11
4	14
5	17
6	20
7	23
8	26
9	29
10	32
11	35

N awalnya = 5
dan selalu ditambah 3

Nilai N pernah = 35
tapi tidak ikut dicetak
karena sudah keluar dari loop

Soal

14e

Susun Algoritma yang ditulis dalam Bahasa C++ untuk mencetak 10 suku pertama deret berikut ini :

5, 8, 13, 20, 29, 40,

Cara-1

Bila algoritma benar, maka tercetak :

5
8
13
20
29
40
53
68
85
104

```
#include<iostream.h>
void main()
{ int I, x;
  X = .... ;
  for( I=1; I<=104; I=I+3 )
  { cout << "\n" << I ;
    I = I + X;
    X = X + .... ;
  }
}
```



I	Tercetak :
5	5
8	8
13	13
20	20
29	29
40	40
53	53
68	68
85	85
104	104
125	

5 8 13 20 29 40 53

+3 +5 +7 +9 +11 +13

Cara-2

```
#include<iostream.h>
void main()
{ int I, N ;
  for( I=1; I<=10; I=I+1 )
  { N = I*I + 4 ;
    cout << "\n" << N ;
  }
}
```

I	N
1	5
2	8
3	13
4	20
5	29
6	40
7	53
8	68
9	85
10	104
11	

Cara-3

```
#include<iostream.h>
void main()
{ int I, N, X;
  N = 5;
  X = 3;
  for( I=1; I<=10; I=I+1 )
  { cout << "\n" << N ;
    N = N + X;
    X = X + 2;
  }
}
```

I	N
1	5
2	8
3	13
4	20
5	29
6	40
7	53
8	68
9	85
10	104
11	125

I	1	2	3	4	5	6	7			
N	5	8	13	20	29	40	53
X	3	5	7	9	11	13				

Soal

15a

Susun program untuk menginput 100 buah bilangan yang merupakan nilai ujian mahasiswa, kemudian cetak nilai tertinggi yang didapat mahasiswa .

Ilustrasi → Bilangan yang

Proses : diinput (N) : 75 72 75 82 64 82 74 82 66 87 72 87 68 ..

Nilai MAX : 75 -- -- 82 -- -- -- -- -- -- -- 87 -- -- -- ..

```
{ int N, I, MAX;
```

```
cin >> N;
```

```
MAX = N; •
```

```
for(I=2; I <= 100; I++) •
```

```
{ cin >> N;
```

```
if (N > MAX) •
```

```
MAX = N ;
```

```
}
```

```
cout << "Tebesar : " << MAX; •
```

```
}
```

Untuk pertama kali, nilai MAX dibuat sama dengan data pertama yang dibaca.

99 kali loop. (dari 2 sampai dengan 100).

Seriap kali loop :
Bila nilai yang baru diinput lebih besar dari MAX, maka nilai MAX diganti dengan nilai yang baru.

Terakhir cetak MAX

Soal**15b**

Susun program untuk menginput 10 buah bilangan yang merupakan nilai ujian mahasiswa, kemudian cetak nilai tertinggi yang didapat mahasiswa , serta mencetak ada berapa orang mahasiswa yang mendapat nilai tertinggi tersebut :

Bila diinput :

15 12 15 10 15 17 25 17 25 13

Seharusnya tercetak :

Nilai terbesar : 25

Jum Mahasiswa : 2

Konsep Proses :

Data : 15 12 15 10 15 17 25 17 25 13

1. Input 15

15

15

1

 Input pertama, isi MAX = N, dan Jum dibuat = 1

N MAX Jum

2. Input 12

12

15

1

 N=12 N tidak > MAX → isi MAX tetap tidak diubah

N MAX Jum N tidak == MAX, → Jum tetap, tidak ditambah

3. Input 15

15

15

2

 N=15 N tidak > MAX → isi MAX tetap, tidak diubah

N MAX Jum Tapi, N == MAX, → Jum ditambah 1

4. Input 10

10

15

2

 N=10 N tidak > MAX → isi MAX tetap tidak diubah

N MAX Jum N tidak == MAX, → Jum tetap, tidak ditambah

5. Input 15

15

15

3

 N=15 N tidak > MAX → isi MAX tetap, tidak diubah

N MAX Jum Tapi, N == MAX, → Jum ditambah 1

6. Input 17

17

17

1

 N=17 N > MAX → isi MAX = N dan Jum = 1

N MAX Jum

Data : 15 12 15 10 15 17 25 17 25 13

17	1
MAX	Jum

7. Input 25

25
N

25
MAX

1
Jum

 N=25 $N > MAX \rightarrow$ isi MAX = N dan Jum = 1

8. Input 17

17
N

25
MAX

1
Jum

 N=17 $N \text{ tidak} > MAX \rightarrow$ isi MAX tetap tidak diubah
 $N \text{ tidak} == MAX, \rightarrow$ Jum tetap, tidak ditambah

9. Input 25

25
N

25
MAX

2
Jum

 N=15 $N \text{ tidak} > MAX \rightarrow$ isi MAX tetap, tidak diubah
Tapi, $N == MAX, \rightarrow$ Jum ditambah 1

10. Input 13

13
N

25
MAX

2
Jum

 N=13 $N \text{ tidak} > MAX \rightarrow$ isi MAX tetap tidak diubah
 $N \text{ tidak} == MAX, \rightarrow$ Jum tetap, tidak ditambah

Data habis :

25
MAX

2
Jum

Jadi dari 12 data yang diinput,
Nilai terbesar = MAX = 25
Dan jumlah mahasiswa yang mendapat nilai
terbesar = Jum = 2

Konsep Proses :



N



MAX



Jum

Input ke : N

MAX menampung nilai terbesar

Jum menampung jumlah mahasiswa yang mempunyai nilai terbesar

Proses : 1. Input pertama, simpan ke **MAX** dan **Jum** dibuat = 1.

2. Input ke **N** 9 kali.

Setiap kali input periksa :

- Bila **N > MAX** Ganti nilai **MAX** dengan nilai N dan nilai **Jum** dibuat ulang = 1.
- Bila N tidak lebih besar dari MAX, Periksa apakah **N == MAX**.

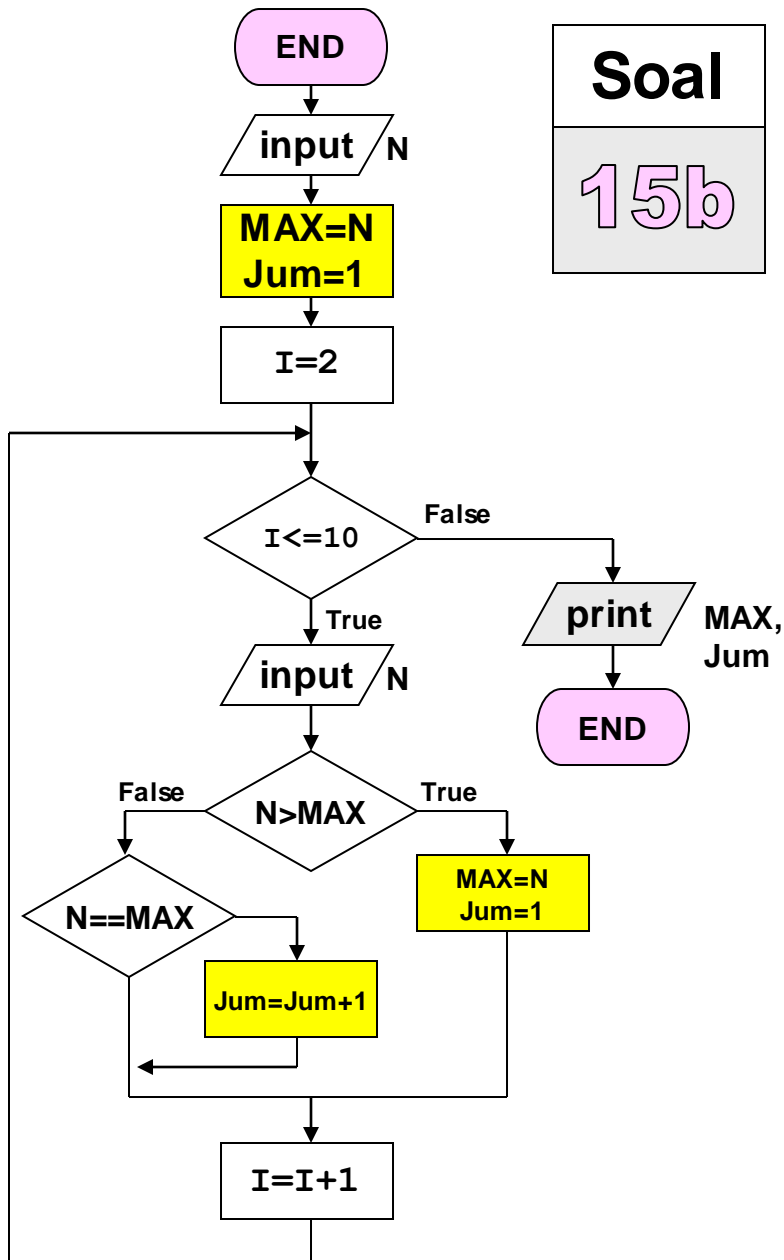
Bila **N == MAX**, **Jum** ditambah 1

3. Setelah selesai 9 kali input, Cetak **MAX** dan **Jum**

Soal

15b

Susun program untuk menginput 10 buah bilangan yang merupakan nilai ujian mahasiswa, kemudian cetak nilai tertinggi yang didapat mahasiswa, serta mencetak ada berapa orang mahasiswa yang mendapat nilai tertinggi tersebut :



```

#include<iostream.h>
void main()
{ int N,I,MAX, Jum;
  cin >> N;
  MAX = N;
  Jum = 1;
  for(I=2; I <= 10; I++)
  { cin >> N;
    if (N > MAX)
    { MAX = N ;
      Jum = 1;
    }
    else
    if (N == MAX)
      Jum++;
  }
  cout << "\nTerbesar : " << MAX;
  cout << "\nJum Mhs : " << Jum;
}
  
```


selesai

Sessi-4

