

Algoritme dan Pemrograman

Kuliah #4

- Struktur kontrol perulangan: while, do ... while, for

Struktur Kontrol Perulangan (*Loop*)

- Program akan **mengulang** satu atau lebih pernyataan berdasarkan kondisi tertentu.
 - Kondisi dinyatakan sebagai ekspresi.
- Kumpulan lebih dari satu pernyataan disebut *compound statement*
 - Diapit kurawal
- Pernyataan yang digunakan disebut *loop statement*.
- Ada tiga bentuk:
 - while
 - do ... while
 - for

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Struktur Kontrol Perulangan: while

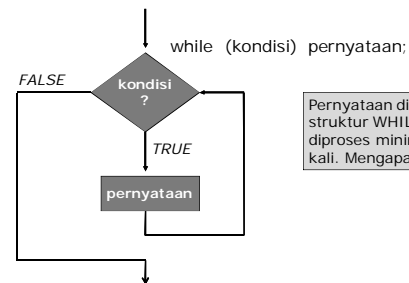
- Lakukan sesuatu selama kondisi bernilai *true*
- Hentikan jika kondisi bernilai *false*.

- Format:

```
while (kondisi)
    simple or compound
    statement;
```

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Struktur Kontrol Perulangan: while



DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Struktur Kontrol Perulangan: while

- Perhatikan potongan program berikut:

```
int i=0;
while (i<5) {
    printf("%d", i);
    i++;
}
```

- Output:
01234

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Struktur Kontrol Perulangan: while

- Apa output potongan program berikut?

```
int i=5;
while (i<10)
    printf("%d", i++);
    printf("%d", i);
```

- Bagaimana dengan potongan program berikut?

```
int i=5;
while (i++<10)
    printf("%d", i);
    printf("%d", i);
```

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Latihan

- Dengan menggunakan pernyataan **while** dan jumlah instruksi sesedikit mungkin, tuliskan program untuk mencetak barisan nilai:

14 8 2 -4 -10 -16

Struktur Kontrol Perulangan: do ... while

- Lakukan sesuatu, dan ulangi selama kondisi bernilai *true*
- Berarti mengerjakan pernyataan di dalamnya terlebih dahulu, baru kemudian memeriksa kondisi yang ada.
 - Pernyataan akan diproses minimal sebanyak berapa kali?
- Format:

```
do {
    pernyataan;
} while (kondisi);
```

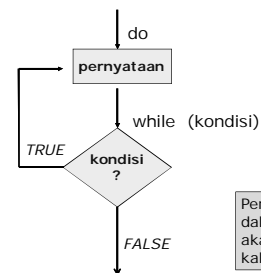
Struktur Kontrol Perulangan: do ... while

```
#include <stdio.h>
int main() {
    int i, n=10;
    i=5;
    do {
        printf("%d\t", i);
        i++;
    } while (i<n);
    return 0;
}
```

Output program tersebut adalah:

5 6 7 8 9

Struktur Kontrol Perulangan: do ... while



Pernyataan (*statements*) di dalam struktur DO ... WHILE akan diproses minimum SATU kali. Mengapa?

Contoh: Program menulis bilangan

```
#include <stdio.h>
int main()
{
    int n;
    scanf("%d", &n);
    do {
        printf("%d", n%10);
        n /= 10;
    } while (n);
    return 0;
}
```

```
#include <stdio.h>
int main()
{
    int n;
    scanf("%d", &n);
    while (n){
        printf("%d", n%10);
        n /= 10;
    }
    return 0;
}
```

Jika diberikan input 789, apa output program tsb? 987

Ubah program agar menggunakan **while**

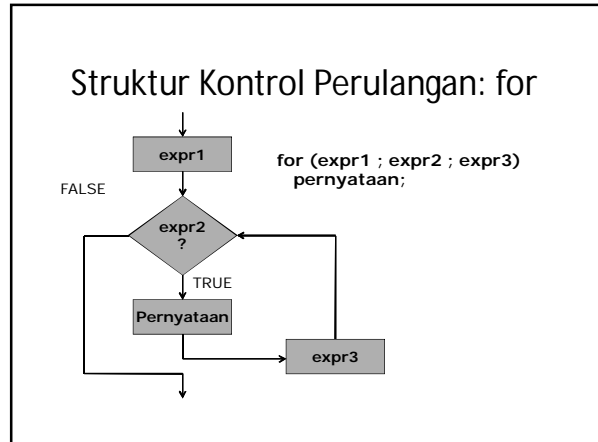
Struktur Kontrol Perulangan: for

- Umumnya digunakan untuk memproses pernyataan secara berulang-ulang dengan jumlah perulangan tertentu
 - Terutama di bahasa pemrograman lain
 - Di C bisa menggantikan **while**
- Contoh:
 - Menuliskan bilangan bulat 1 sampai dengan 5
 - Membaca n bilangan bulat.

Struktur Kontrol Perulangan: for

- Bentuk umum


```
for (expr1; expr2; expr3) {
    s1;
    s2;
    ...
}
```
- Proses yang dilakukan mengikuti alur berikut:
 - expr1 dieksekusi hanya satu kali sebelum mulai looping.
 - expr2 merupakan ekspresi boolean. Jika tidak dituliskan, diasumsikan bernilai true.
 - Jika expr2 bernilai false, maka proses looping dihentikan.
 - Jika expr2 bernilai true, maka s1, s2, ... dieksekusi.
 - Setelah eksekusi s1, s2, ..., maka expr3 dieksekusi.
- expr1, expr2, dan expr3 bersifat opsional (dapat dituliskan, juga dapat tidak dituliskan).



for: Contoh #1

```
#include <stdio.h>
int main(){
    int i, n = 10;

    for (i=5; i<n; i++) {
        printf("%d", i);
    }
    return 0;
}
```

Apa output program tersebut?
(trace dengan diagram alir for) 56789

for: Contoh #2 (backward loop)

```
#include <stdio.h>
int main(){
    int i=3;

    for ( ; i; i-- ) {
        printf("%d", i);
    }
    return 0;
}
```

Apa output program tersebut? 321
Bagaimana jika baris for menjadi for(; i--;)? 210

for: Contoh #3 (backward loop)

```
#include <stdio.h>
int main(){
    int i=5, n=10, sum=0;

    for ( ; i<=n; sum+=i++);

    printf("%d\n", sum);
    return 0;
}
```

Apa output program tersebut? 45
Apa yang dilakukannya?
Bagaimana jika menggunakan for(i++; i<n; sum+=i);?

for: Contoh #4

```
#include <stdio.h>
int main()
{
    int i, n, bil, sum=0;
    scanf("%d", &n);
    for (i=1; i<=n; i++) {
        scanf("%d", &bil);
        sum+=bil;
    }
    printf("%d\n", sum);
    return 0;
}
```

Apa output program jika diberi input 3 44 7 9? 60
Apa yang dilakukan program tersebut?

Output?

```

...
int a, b=4;
for ( ; a=b, b-- ; )
    printf("%d %d\n",a,b);
...

```

4 3
 3 2
 2 1
 1 0

Struktur Kontrol Perulangan: nested for

- Pernyataan for yang bersarang, artinya terdapat pernyataan for lainnya di dalam pernyataan for.
 - Dapat juga dilakukan untuk while dan do... while
- Contoh, apa output potongan program berikut?

```

int i, j;
for (i=1; i<=3; i++) {
    for (j=1; j<=2; j++) {
        printf("**");
    }
    printf("\n");
}

```

Struktur Kontrol Perulangan: nested for

- Misalkan ingin menuliskan bilangan 1, 2, ... dalam bentuk matriks ukuran 3x4. Setiap baris hanya diakhiri oleh newline.

```

#include <stdio.h>

int main() {
    int i, j, c=1;
    for (i=1; i<=3; i++)
        for (j=1; j<=4; j++) {
            printf("%d", c++);
            if (j!=4) printf(" ");
            else printf("\n");
        }
    return 0;
}

```

Nested for:

Menggunakan variabel sebelumnya

- Perhatikan contoh berikut. Struktur for untuk variabel *j* menggunakan variabel *i*.
- Perhatikan juga posisi inisialisasi sum
- Apa outputnya? Apa yang dilakukan?

```

int i, j, sum;
for (i=1; i<=4; i++) {
    sum=0;
    for (j=1; j<=i; j++)
        sum+=j;
    printf("%d %d\n", i, sum);
}

```

Nested FOR:
Menggunakan variabel sebelumnya

- Apa output potongan program berikut?

```

int i, j, n=5;
for (i=1; i<=n; i++) {
    for (j=i; j<=n; j++)
        printf("*");
    printf("\n");
}

```


 **
 *

Latihan #1

- Berapa kali string IPB dicetak oleh program berikut?

```

#include <stdio.h>

int main() {
    int i;
    for (i=1; i<=20; i++) {
        if (i=5)
            printf ("IPB\n");
    }
    return 0;
}

```

Latihan #2

- Buat program untuk membaca sebuah bilangan bulat n , dan menampilkan output seperti dalam contoh. Setiap baris output hanya diakhiri dengan *newline*.
- Contoh input: 5
- Contoh output:


```
*
**
***
****
*****
```

```
int main() {
    int i, j, n=5;
    for (i=1; i<=n; i++) {
        for (j=1; j<=i; j++)
            printf("*");
        printf("\n");
    }
    return 0;
}
```

Latihan #4

- Apa output potongan program berikut jika diberikan input data 16 24? 8
- Apa yang dilakukan?

```
int a, b, t;
scanf("%d %d", &a, &b);
while (b) {
    t = a%b;
    a = b;
    b = t;
}
printf("%d\n", a);
```

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Latihan #5

- Apa output potongan program berikut jika diberikan input data -5 77 3 -1 -99? 4
- Apa yang dilakukan?

```
int bil, c=0;
scanf("%d", &bil);
while (bil!=-99) {
    c++;
    scanf("%d", &bil);
}
printf("%d\n", c);
```

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Latihan #6

- Apa output potongan program berikut jika diberikan input 7251? 1527
- Apa yang dilakukan?

```
int bil, t=0;
scanf("%d", &bil);
while (bil) {
    t = (t*10) + (bil%10);
    bil/=10;
}
printf("%d\n", t);
```

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

Latihan #7

Suatu sensor dipasang pada tanaman untuk mengatur suplai air. Sensor secara periodik mengukur kandungan air tanah di sekitar tanaman. Jika terdeteksi data kandungan air tanah kurang dari 4.5 satuan, alat akan menyuplai air sebanyak 0.12 liter. Demikian seterusnya sampai diperoleh data kandungan air tanah lebih besar atau samadengan 4.5 satuan dan alat akan berhenti sementara. Buat program C untuk menghitung total volume air yang disuplai sampai alat tersebut berhenti sementara. Output ditulis dalam dua desimal di belakang titik.

Contoh input : 1.2 4.2 4.3 4.4 5.2
Contoh output : 0.48

Contoh input : 4.9
Contoh output : 0.00

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

```
float bil=0, jum=0;
while (bil<4.5) {
    jum += 0.12;
    scanf("%d", &bil);
}
printf("%d\n", jum);
```

Latihan #8

Seorang anak suka bermain dengan bilangan bulat. Jika diberikan bilangan bulat lebih besar atau samadengan 100, dia hanya akan mengambil digit paling kiri dan paling kanan untuk membentuk suatu bilangan baru dengan dua digit. Misalkan diberikan nilai 7895, dia membentuk bilangan baru yaitu 75. Selanjutnya, dia menuliskan sisa pembagian bilangan baru tersebut dengan 6 di kertas. Jika bilangan bulat yang diberikan kurang dari 100, maka dia tidak melakukan apa-apa. Buat program C untuk mendapatkan nilai seperti yang dituliskan anak tadi.

Contoh input : 7895

Contoh input : 37

Contoh output : 3

Contoh output :

DEPARTEMEN ILMU
KOMPUTER INSTITUT
PERTANIAN BOGOR

```
int kiri,kanan,bil,jum, i, sisa;
scanf("%d", &bil);
```

```
kanan = bil % 10;
```

```
while (bil>10) {
    bil = bil/10;
}
```

```
kiri = bil * 10;
```

```
bil = kiri + kanan;
```

```
bil=bil%6;
```

```
printf("%d\n", bil);
```

Latihan #9

```
int a=3, b=0;
while (b++<a) printf("%d", b);           1 2 3
int a=3, b=0;
while (++b<a) printf("%d", b);           1 2
int a=5, b=0;
while (b++<a) printf("%d", b++);         1 3 5
int a=3, b=0;
while (b=a--) printf("%d %d\n", a,b);    2 3
int a=3, b=0;
while (b=--a) printf("%d %d\n", a,b);    0 1
int a=5, b, c;
while (b=--a) printf("%d %d\n", a,b);    2 2
int a=5, b, c;
while (c=--a, b=a--) printf("%d %d %d\n", 3 4 4
a,b,c);                                  1 2 2
```