

## MODUL 3 STATEMENT PERULANGAN

Perulangan adalah sekumpulan instruksi yang dieksekusi secara berulang-ulang hingga kondisi pemberhentian dipenuhi. Macam-macam perulangan pada bahasa C :

### 1. Perulangan berhingga (*Counter Controller Repetition*)

Perulangan berhingga yaitu suatu perulangan dimana nilai awal dan akhirnya ditentukan oleh programmer.

Syntax :

```
For (inisialisasi; syarat_perulangan; peningkatan/penurunan) {  
  
    statement_1;  
  
    statement_2;  
  
    statement_N;  
  
}
```

Keterangan:

- “inisialisasi” digunakan untuk memberikan nilai awal perulangan
- “syarat perulangan” memegang kontrol perulangan atau sebagai batas perulangan yang akan menentukan perulangan akan diteruskan atau dihentikan
- “peningkatan/penurunan” digunakan untuk mengatur kenaikan atau penurunan perulangan

Contoh program perulangan berhingga :

```
1. #include<stdio.h>  
2. Main(){  
3.     Int i;  
4.     For(i=0; i,10; i++) {  
5.         printf(“%d. Sedang belajar C\n”,i);  
6.     }  
7. }
```

### 2. Perulangan Tak Berhingga (*Sentinel Controller repetition*)

Perulangan tak berhingga merupakan perulangan dimana jumlah pengulangannya tidak dapat ditentukan sehingga pengulangan akan terus dilakukan selama kondisi bernilai benar atau salah.

Ada beberapa pernyataan dalam perulangan tak berhingga, yaitu :

#### a. Pernyataan *while*

Yaitu selama kondisi bernilai benar maka perulangan akan terus dilakukan dan jika salah maka akan dihentikan.

Syntax :

```
while (kondisi) {  
    statement_1;  
    statement_2;  
    statement_N;  
}
```

Contoh program perulangan tak berhingga “while” :

```
1. #include<stdio.h>  
2. int main(){  
3.     int bil, jumlah, n;  
4.     float rerata;  
5.     jumlah=0;  
6.     n=0;  
7.     printf(“masukkan bilangan = “);  
8.     scanf(“%d”,&bil);  
9.     while(bil !=0){  
10.         jumlah+=bil;  
11.         n++;  
12.         scanf(“%d”,&bil);  
13.     }  
14.     printf(“\n”);  
15.     rerata=jumlah/n;  
16.  
17.     printf(“rerata = %.1f\n”, rerata);  
18.     return 0;  
19. }
```

b. Pernyataan *do-while*

Yaitu bentuk perulangan yang melakukan perulangan terlebih dahulu dan pengecekan dilakukan diakhir pernyataan dan perulangan akan terus dilakukan selama kondisi salah.

Syntax :

```
do {  
    statement_1;  
    statement_2;  
    statement_N;  
} while(kondisi);
```

Contoh program pernyataan *do-while* :

```
1. #include<stdio.h>
2. main(){
3.     int bil=2;
4.     do{
5.         printf("%d\t", bil);
6.         bil+=2;
7.     } while (bil <= 9);
8.     printf("\n");
9. }
```

c. Pernyataan Goto

Yaitu intruksi untuk mengarahkan eksekusi program ke pernyataan yang diawali dengan suatu label yang menjadi tanda pengenal dan diikuti oleh tanda titik dua (:)

Syntax :

```
Nama_label :
    Statement_1;
    Statement_2;
    Statement_N;
Goto Nama_label;
```

Contoh program pernyataan Goto :

```
1. #include<stdio.h>
2. main(){
3.     int a,b,c;
4.     kembali;
5.     print("\n masukkan angka : ");
6.     scanf("%d",&a);
7.     b=a%2;
8.     printf("%d mod 2 = %d\n",a,b);
9.     printf("hitung kembali ? [1/0] : ");
10.    scanf("%d",&c);
11.    if(c==1)
12.        goto kembali;
13. }
```

d. Pernyataan break

Yaitu untuk keluar dari suatu struktur switch atau keluar dari perulangan (*for, while, do-while*)

Contoh program pernyataan break :

```
1. #include<stdio.h>
2. main(){
3.     int bil=1;
4.     do{
5.         if(bil >=6)
6.             break;
7.         printf("%i",bil);
8.     } while(bil++);
9.     printf("\n");
10. }
```

e. Pernyataan continue

Yaitu untuk mengarahkan eksekusi ke iterasi (proses) berikutnya pada loop yang sama tanpa menjalankan sisa perintah yang ada di loop

Contoh program pernyataan continue :

```
1. #include<stdio.h>
2. main(){
3.     int bil;
4.     for(bil=0; bil<10; bil++){
5.         if(bil==6)
6.             continue;
7.         printf("%i",bil);
8.     }
9.     printf("\n");
10. }
```

- Program menentukan bilangan absolute dari bilangan yang diinput

tes abs.cpp

```

1  #include<stdio.h>
2  #include<conio.h>
3  main()
4  {
5      int a,abs;
6      printf("masukkan angka : ");
7      scanf("%d",&a);
8      if (a<0)
9          abs = -a;
10     printf("nilai absolute dari %d adalah %d\n\n",a,abs);
11     getch ();
12 }

```

C:\Users\Lenovo\Desktop\tes abs.exe

```

masukkan angka : -9
nilai absolute dari -9 adalah 9

```

- Program bilangan terbesar dan terkecil dari 3 bilangan yang diinput

bilangan terbesar dan terkecil jadi.cpp

```

1  # include <stdio.h>
2  # include <conio.h>
3
4  main ()
5  {
6      int data [50];
7      int a,b;
8      int max,min;
9
10     printf("Banyak Bilangan: ");scanf ("%d",&a);
11
12     for (b=1;b<=a;b++)
13     {
14         printf("\nbilangan ke-%d: ",b);scanf("%d",&data [b]);
15     }
16
17     max=data[1];
18     min=data[1];
19     for (b=1;b<=a;b++)
20     {
21         if (data[b]>=max)
22         {
23             max=data[b];
24         }
25         else if (data[b]<=min)
26         {
27             min=data[b];
28         }
29     }
30     printf("\n\n nilai tertinggi adalah \t:%d", max);
31     printf("\n\n nilai terendah adalah \t:%d", min);
32     getch();
33 }

```

D:\codingan devc++\bilanga...

```

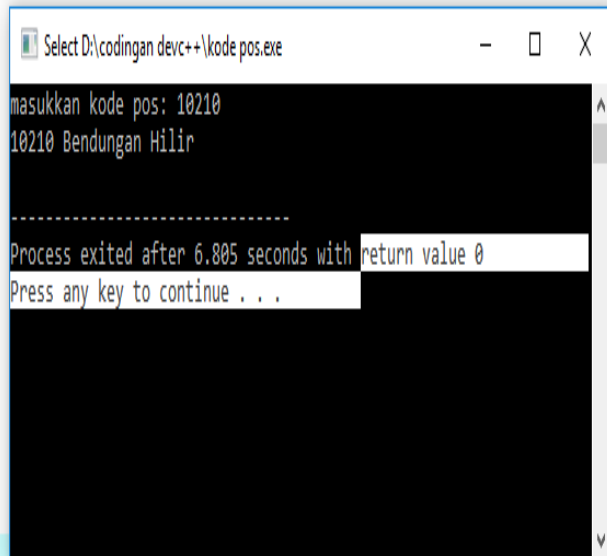
Banyak Bilangan: 3
bilangan ke-1: 54
bilangan ke-2: 34
bilangan ke-3: 12

nilai tertinggi adalah      :54
nilai terendah adalah      :12

```

- Program menentukan kode pos untuk daerah tanah abang

```
1 #include<stdio.h>
2 int main()
3 {
4     int angka;
5     printf("masukkan kode pos: "); scanf("%d",&angka);
6     switch (angka) {
7         case 10210: printf("%i Bendungan Hilir\n", angka); break;
8         case 10220: printf("%i Karet Tengsin\n", angka); break;
9         case 10230: printf("%i Kebon Melati\n", angka); break;
10        case 10240: printf("%i Kebon Kacang\n", angka); break;
11        case 10250: printf("%i Kampung Bali\n", angka); break;
12        case 10260: printf("%i Petamburan\n", angka); break;
13        case 10270: printf("%i Gelora\n", angka); break;
14    }
15 }
16
```



```
Select D:\codingan devc++\kode pos.exe
masukkan kode pos: 10210
10210 Bendungan Hilir

-----
Process exited after 6.805 seconds with return value 0
Press any key to continue . . .
```