

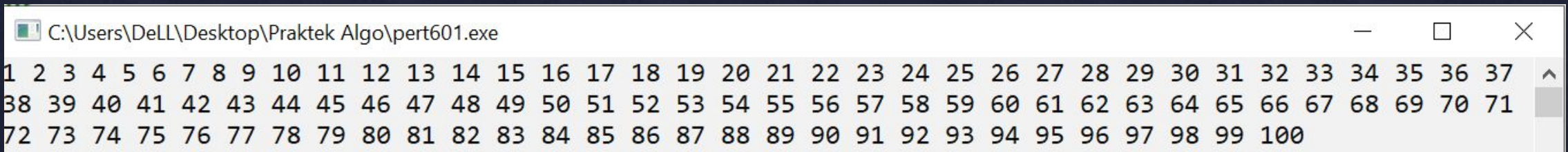
Struktur Perlulangan

Apakah anda ingin melakukan Transaksi Lain?



Pengertian

- ❖ Perulangan (atau dalam bahasa inggris disebut dengan *loop*) adalah instruksi program yang bertujuan untuk mengulang beberapa baris perintah.
- ❖ Fungsi paling sederhana yaitu untuk mempermudah melakukan suatu proses yang berulang-ulang, seperti mencetak angka dari 1 – 100



A screenshot of a Windows command prompt window. The title bar shows the file path: C:\Users\DeLL\Desktop\Praktek Algo\pert601.exe. The window contains a list of numbers from 1 to 100, arranged in three rows. The first row contains numbers 1 through 37, the second row contains 38 through 71, and the third row contains 72 through 100. A vertical scrollbar is visible on the right side of the text area.

```
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37
38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71
72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100
```


Jenis Perulangan

- ◆ Jenis – jenis perulangan dalam bahasa C++ terdiri dari :
 - ▢ Perulangan for
 - ▢ Perulangan nested – for
 - ▢ Perulangan tak terhingga
 - ▢ Pernyataan go to
 - ▢ Perulangan while
 - ▢ Perulangan do – while
 - ▢ Pernyataan break
 - ▢ Perulangan continue

Struktur perulangan

- ◆ Struktur perulangan secara umum terdiri dari dua bagian, yaitu:
 1. Kondisi perulangan, yaitu ekspresi Boolean yang harus dipenuhi untuk melaksanakan pengulangan
 2. Body pengulangan, yaitu bagian algoritma yang diulang

Struktur perulangan

- ◆ Penulisan struktur perulangan secara umum :

<inisialisasi>

badan pengulangan

<terminasi>

Perulangan for

- ◆ Perulangan for digunakan untuk menghasilkan pengulangan sejumlah kali yang telah dispesifikasikan.
- ◆ Jumlah pengulangan dapat ditentukan sebelum dieksekusi.
- ◆ Bentuk umum pernyataan for :

```
For(inisialisasi ; syarat pengulangan; pengubah nilai pencacah)
```

- ◆ Bila pernyataan di dalam for lebih dari satu maka pernyataan tersebut harus diletakan seperti dibawah ini :

```
For(inisialisasi : syarat pengulangan; pengubah nilai pencacah)
{
    perintah;
    perintah;
}
```

Perulangan for (lanjutan)

Kegunaan dari masing – masing argument for adalah :

- ◆ Inisialisasi:
 - ▢ Merupakan bagian untuk memberikan nilai awal untuk variabel – variabel tertentu
- ◆ Syarat pengulangan :
 - ▢ Memegang kontrol terhadap pengulangan, karena bagian ini yang akan menentukan suatu perulangan diteruskan atau dihentikan
 - ▢ Pengubah nilai pencacah
 - ▢ Mengatur kenaikan atau penurunan nilai pencacah

Perulangan for (lanjutan)

Contoh kasus perulangan for :

Program ini untuk mencetak bilangan dari 1 hingga 10 secara menaik dan secara menurun

```
#include <iostream>
using namespace std;

int main(){

    for (int a=1;a<=10;a++){
        cout<<a<<" Uvers"<<endl;
    }

}
```

```
#include <iostream>
using namespace std;

int main(){

    for (int a=10;a>=1;a--){
        cout<<a<<" Uvers"<<endl;
    }

}
```

Perulangan Nested – for

- ❖ Perulangan nested for adalah suatu perulangan for di dalam perulangan for yang lainnya
- ❖ Didalam penggunaan nested-for, perulangan yang didalam terlebih dahulu dihitung hingga selesai, kemudian perulangan yang diluar diselesaikan.
- ❖ Bentuk umum pernyataan Nested for, sebagai berikut :

```
for ( inisialisasi; syarat pengulangan; pengubah nilai pencacah )  
{  
    for ( inisialisasi; syarat pengulangan; pengubah nilai pencacah )  
    {  
        pernyataan / perintah;  
    }  
}
```

Perulangan Nested - for

- ◆ Contoh perulangan nested-for dengan bahasa C++

```
#include <iostream>
using namespace std;

int main(){
    for (int a=1; a<=5; a++){
        for (int b=a; b<=5; b++){
            cout<<a<<" ";
        }
        cout<<endl;
    }
}
```


Perulangan Tak Terhingga

- ❖ Perulangan tak terhingga merupakan perulangan (loop) yang tak pernah berhenti atau terus mengulang perintah
- ❖ Hal ini sering terjadi, ketika terjadi kesalahan dalam penanganan kondisi yang dipakai untuk keluaran.

Perulangan Tak Terhingga(lanjutan)

- ◆ Contoh bahasa c++
- ◆ Ket : perintah ini akan membuat perulangan tidak berhenti dikarenakan pada bilangan++ seharusnya ditulis bilangan- - dan jika ingin memberhentikan perulangan ini dengan cara tekan CTRL – PAUSE atau CTRL – Break

```
#include <iostream>
using namespace std;

int main(){
    for (int a=10;a>=1;a++){
        cout<<a;
    }
}
```

Pernyataan goto

```
#include <iostream>
using namespace std;

int main(){
    int u;
    char lagi;
    atas:
    cout<<"Masukkan Umur: ";
    cin>>u;

    cout<<"Umur anda adalah: "<<u<<endl;
    cout<<"Ulang lagi (y/t): ";
    cin>>lagi;

    if (lagi == 'y'){
        goto atas;
    }
}
```

- ❖ Pernyataan goto merupakan instruksi untuk mengarahkan eksekusi program ke pernyataan yang diawali dengan suatu label.
- ❖ Label merupakan suatu pengenalan yang diikuti dengan tanda (:) titik dua.
- ❖ Bentuk penulisan goto, adalah sebagai berikut :

```
goto label;
```


Perulangan While

- ◆ Perulangan while merupakan instruksi perulangan yang mirip dengan perulangan for.
- ◆ Bentuk perulangan while di tentukan oleh syarat tertentu, yaitu perulangan akan terus dilakukan selama syarat tersebut terpenuhi

Perulangan While(lanjutan)

- ◆ Bentuk umum perulangan while 1 kondisi :

```
While(syarat)  
    Pernyataan/perintah;
```

- ◆ Bentuk umum perulangan while banyak kondisi :

```
While(syarat)  
{  
    Pernyataan/perintah;  
    pernyataan/perintah;  
}
```

Perulangan While(lanjutan)

- ◆ Contoh perulangan while bahasa C++

```
#include <iostream>
using namespace std;

int main(){
    int c = 1;
    while (c <= 10){
        cout<<c<<endl;
        c++;
    }
}
```


Perulangan do - while

- ◆ Perulangan do - while merupakan perulangan yang melaksanakan perulangan terlebih dahulu dan pengujian perulangan dilakukan di belakang.
- ◆ Bentuk umum perintah perulangan do - while :

```
Do  
    Pernyataan/perintah;  
While(syarat)
```

- ◆ Bentuk umum perintah perulangan do - while banyak kondisi

```
Do  
{  
    Pernyataan/perintah;  
    Pernyataan/perintah;  
}  
While(syarat)
```

Perulangan do – while(lanjutan)

- ◆ Contoh pada bahasa C++

```
#include <iostream>
using namespace std;

int main(){
    int c = 1;
    do {
        cout<<c<<endl;
        c++;
    } while (c<=10);
}
```

Pernyataan Break

- ◆ Pernyataan break sebenarnya sudah dijelaskan pada penyeleksian perintah switch
- ◆ Pernyataan break berfungsi sebagai keluar dari struktur switch.
- ◆ Pada perulangan pernyataan break berfungsi untuk keluar dari perulangan (*for, while, do-while*)
- ◆ Jika perintah break dikerjakan, maka eksekusi akan dilanjutkan ke pernyataan yang terletak sesudah akhir perintah perulangan (*looping*)

Pernyataan Break

- ◆ Contoh penggunaan bahasa C++

```
#include <iostream>
using namespace std;

int main(){
    int c = 1;
    do {
        if (c >=6)
        {
            break;
        }
        cout<<c<<endl;
    } while (c++);
}
```

Pernyataan Continue

- ◆ Pernyataan continue digunakan untuk mengarahkan eksekusi ke iterasi (proses) berikutnya pada perulangan yang sama dengan kata lain mengembalikan proses yang sedang dilaksanakan ke awal perulangan lagi, tanpa menjalankan sisa perintah dalam perulangan

Pernyataan Continue

- ◆ Contoh pernyataan continue

```
#include <iostream>
using namespace std;

int main(){
    for (int a=1; a<=10;a++){
        if (a == 6){
            continue;
        }
        cout<<a<<" ";
    }
}
```




RAKORNAS
APTIKOM
2020



**"Mewujudkan SDM Unggul
melalui Kolaborasi
Smart Campus dengan
Stakeholders dalam
Menerapkan **Merdeka
Belajar dan Kampus
Merdeka**"**

RAKORNAS APTIKOM
2020

#AptikomJaya
#AptikomUntukIndonesia
#GabungRakornasAptikom2020

Terima Kasih

