

# Pemrograman dengan C++ Builder

## 6.1. Perulangan

Dari bahasa pemrograman yang pernah anda pelajari sebelumnya, tentu anda pernah menemukan penggunaan perulangan (Looping). Proses perulangan ini merupakan proses dimana komputer akan mengeksekusi lebih dari satu statement secara berulang-ulang sesuai dengan kebutuhan.

Perulangan dengan menggunakan Statement for digunakan untuk mengulang statement atau satu blok statement berulang kali, sejumlah yang ditentukan. Perulangan dengan statement for ini dapat berbentuk perulangan:

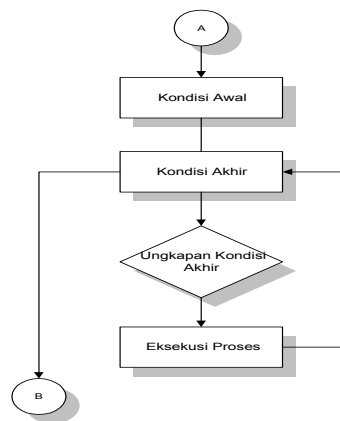
1. Positif
2. Bersarang (Nested)

Aturan penulisan perulangan for adalah:

for (pernyataan kondisi awal, pernyataan kondisi akhir, pernyataan proses)

```
{  
  Statement  
}
```

Bentuk Diagram For

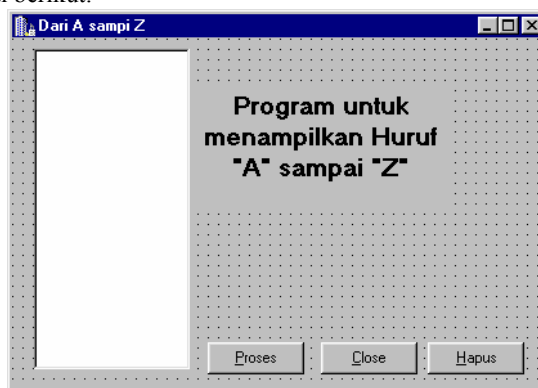


Gambar 6.1. Diagram For

## 6.2. Perulangan Positif

Perulangan Positif merupakan perulangan dengan penghitung (counter) dari kecil ke besar dengan penambahan Positif. Sebagai contoh di bawah ini akan kita coba untuk membuat sebuah program yang menampilkan Abjad yang dimulai dari Huruf "A" sampai dengan Huruf "Z".

1. Buat Form seperti berikut:



Gambar 6.2 Desain Form Perulangan Positif

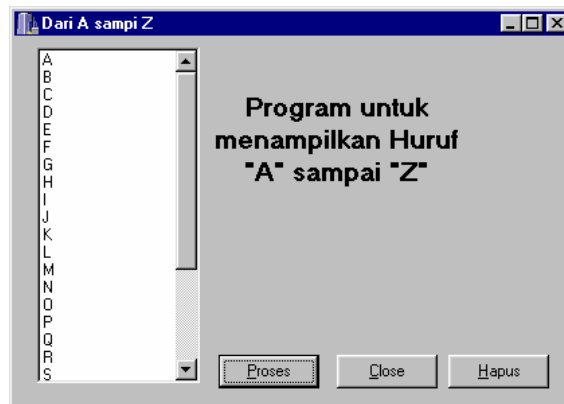
## Pemrograman dengan C++ Builder

2. Kemudian anda masukan kode berikut pada Tombol Proses  

```
void __fastcall TForm1::Button1Click(TObject *Sender)
{
    char abjad;
    for (abjad='A'; abjad <='Z'; abjad++)
        ListBox1->Items->Add(abjad);
}
```
3. Kemudian anda masukan kode berikut ke dalam tombol Close  

```
void __fastcall TForm1::Button2Click(TObject *Sender)
{
    Close();
}
```
4. Kemudian anda masukan kode berikut ke dalam tombol Hapus  

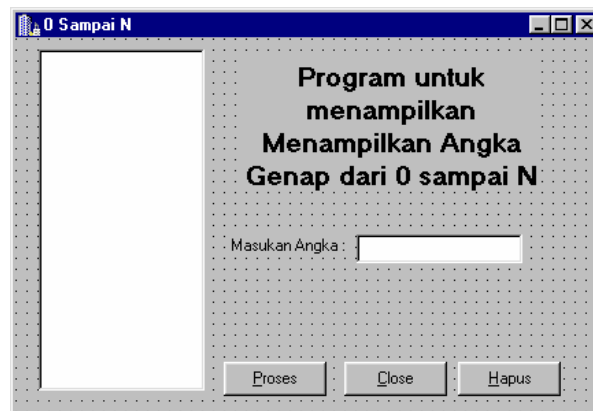
```
void __fastcall TForm1::Button3Click(TObject *Sender)
{
    ListBox1->Items->Clear();
}
```
5. Coba anda jalankan Program tersebut.
6. Bila tidak ada kesalahan anda akan mendapatkan hasil seperti gambar di bawah ini.



Gambar 6.3. Ouput Program Perulangan Positif

Berikut ini contoh Program lainnya yang menampilkan semua angka genap dari 0 sampai yang paling paling besar. Ukuran yang paling besar diambil dari kontrol Edit1. Berikut Desain Form yang diinginkan.

1. Buat form seperti berikut



Gambar 6.4. Desain Form Menampilkan Angka Genap

## Pemrograman dengan C++ Builder

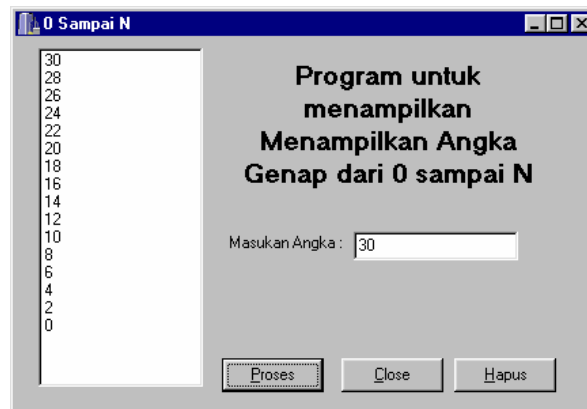
2. Masukkan Kode Berikut ke dalam Tombol Proses

```
void __fastcall TForm2::Button1Click(TObject *Sender)
{
    int n ;
    n=StrToInt(Edit1->Text);
    if (n%2)
        n--;
    for (; n >=0; n-=2)
        ListBox1->Items->Add(n);
}
```

3. Masukkan kode berikutnya ke dalam tombol Close dan Tombol Hapus

```
void __fastcall TForm2::Button2Click(TObject *Sender)
{
    Close();
}
//-----
void __fastcall TForm2::Button3Click(TObject *Sender)
{
    ListBox1->Items->Clear();
}
```

4. Berikut ini Output yang dihasilkan bila tidak ada kesalahan.



Gambar 6.5. Output Program Menampilkan Angka Genap

### 6.3. Perulangan Bersarang

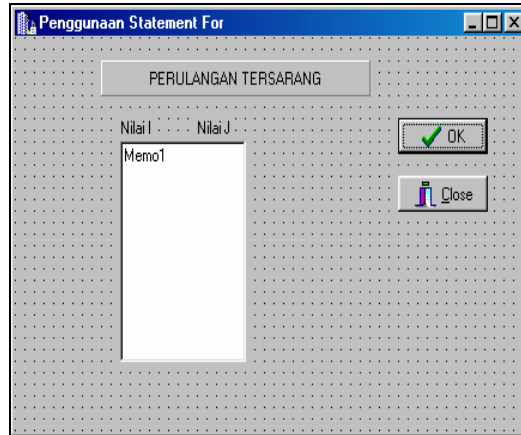
Perulangan bersarang atau (Nested Loop) merupakan perulangan yang berada didalam perulangan yang lain. Perulangan yang lebih dulu diproses adalah perulangan yang terletak di bagian dalam, setelah habis kemudian perulangan yang lebih luar akan diproses, sampai seluruh kondisi terpenuhi.

Aturan Penulisan:

```
for(Variabel-1 nilaiawal;Variabel-1 nilaiakhir;Variabel1-penambahan)
{
    for(Variabel-2 nilaiawal;Variabel-2 nilaiakhir;Variabel2-penambahan)
    {
        Statement
    }
}
```

Berikut adalah contoh program menggunakan perulangan bersarang

1. Buat Form berikut:

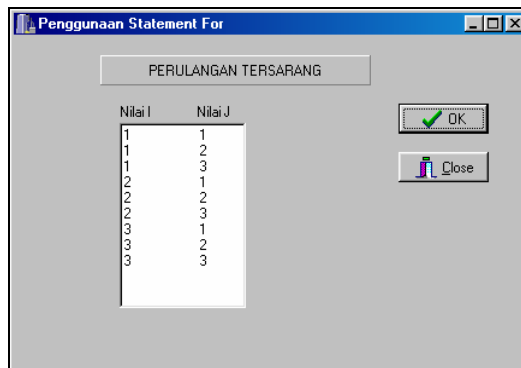


Gambar 6.6. Desain Form Perulangan Bersarang

2. Pada tombol Ok, masukan kode berikut:

```
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
{
    int i,j;
    Memo1->Lines->Clear();
    for (i=1;i<=3;i++)
    {
        for (j=1;j<=3;j++)
        {
            Memo1->Lines->Add(IntToStr(i)+ " +
            IntToStr(j));
        }
    }
}
```

3. Jalankan program tersebut, kemudian tekan tombol Ok, maka akan ditampilkan output seperti berikut:



Gambar 6.7. Output Program Perulangan Bersarang

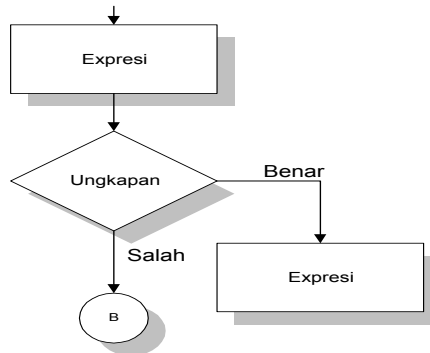
### 6.4. Pernyataan While

Pernyataan while merupakan salah satu pernyataan yang berguna untuk memproses suatu pernyataan atau beberapa pernyataan beberapa kali.

Bentuk pernyataan ini adalah :

```
While(ungkapan)
    Pernyataan;
```

Bentuk Diagram While



Gambar 6.8. Diagram While

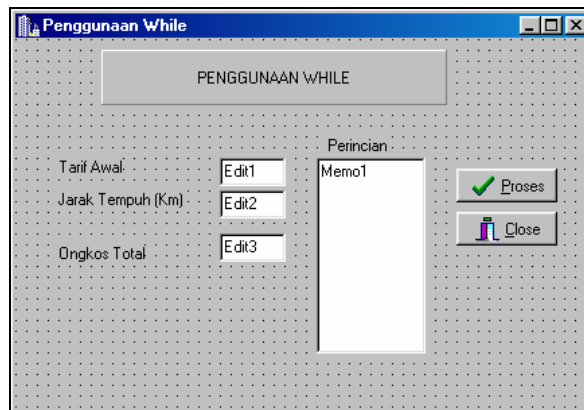
Bagian pernyataan tertentu saja bisa berupa pernyataan majemuk sehingga bentuknya menjadi seperti berikut:

```
While(ungkapan)
{
    pernyataan1;
    pernyataan2;
    ....
    pernyataanN;
}
```

Bagian pernyataan yang mengikuti while akan dieksekusi selama ungkapan pada while bernilai benar (tidak sama dengan nol). Perlu diketahui pengujian terhadap ungkapan while dilakukan sebelum bagian pernyataan. Oleh karena itu ada kemungkinan bagian pernyataan pada while tidak dijalankan sama sekali, yaitu kalau kondisi yang pertama kali bernilai salah.

Contoh Program:

1. Buat Form seperti berikut:



Gambar 6.9. Desain Form While

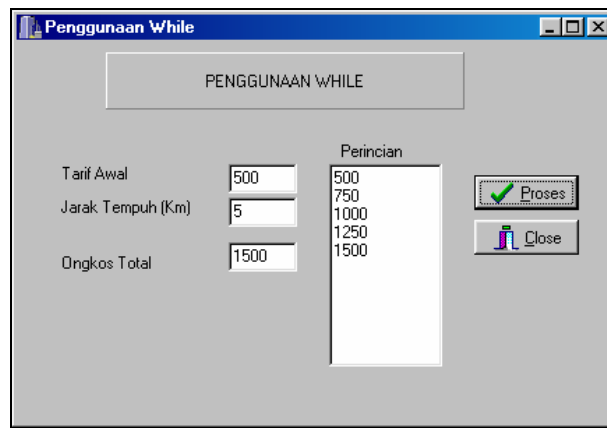
2. Masukkan Kode berikut pada tombol proses  

```
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
{
    int awal, jarak, total, i=0, tarifKm=250;
```

## Pemrograman dengan C++ Builder

```
Memo1->Lines->Clear();
awal=StrToInt(Edit1->Text);
jarak=StrToInt(Edit2->Text);
jarak=jarak-1;
while(i<=jarak)
{
total=awal+(tarifKm*i);
Memo1->Lines->Add(IntToStr(total));
i++;
}
Edit3->Text=IntToStr(total);
}
```

3. Jalankan program tersebut, maka akan menghasilkan output seperti berikut:



Gambar 6.10. Output Program While

### 6.5. Pernyataan do-while

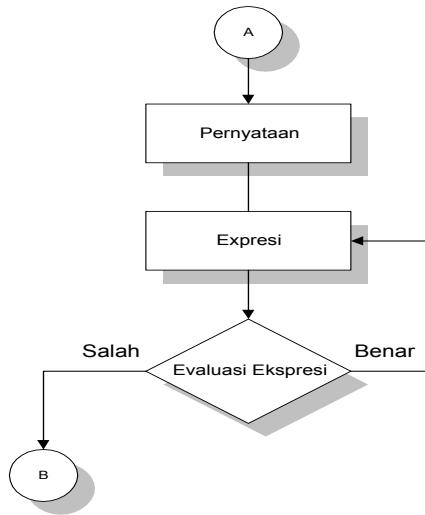
Pernyataan do-while juga berguna untuk menanggulangi proses berulang, bentuk penulisannya adalah seperti berikut:

```
do
{
pernyataan1;
pernyataan2;
.....
pernyataanN
} while (ungkapan)
```

Bagian pernyataan1 hingga pernyataanN dijalankan secara berulang sampai ungkapan bernilai salah. Namun berbeda dengan while, pengujian ungkapan dilakukan dibelakang (setelah bagian pernyataan)

Perulangan ini merupakan teknik pengembangan dari while. Pada jenis perulangan ini, pernyataan akan dijalankan lebih dulu sebelum evaluasi nilai eskpresi.

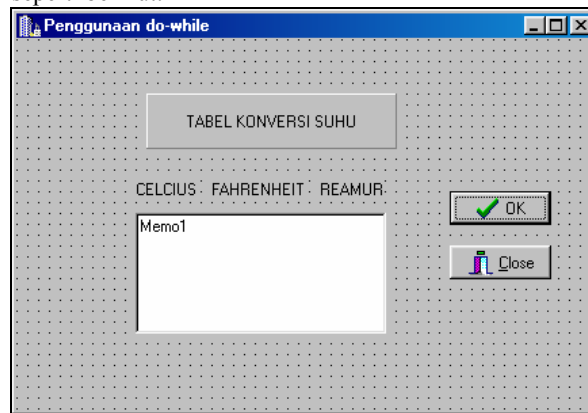
Bentuk Diagram do-while



Gambar 6.11. Diagram do-while

Contoh:

1. Buat Form seperti berikut:

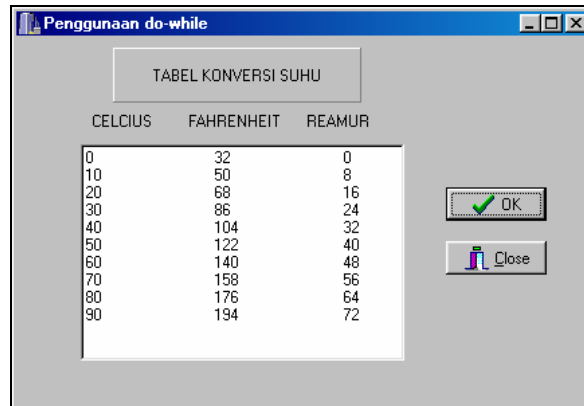


Gambar 6.12. Desain Form do-while

2. Masukan kode berikut pada tombol OK  

```
void __fastcall TForm1::BitBtn1Click(TObject *Sender)
{
    int Celcius=0;
    float Fahren,Reamur;
    Memo1->Lines->Clear();
    do
    {
        Fahren=9./6. * Celcius + 32;
        Reamur=4./6.*Celcius;
        Memo1->Lines->Add(IntToStr(Celcius)+"\t\t"
        +FloatToStr(Fahren)+"\t\t"+FloatToStr(Reamur));
        Celcius+=10;
    }while(Celcius<100);
}
```

3. Jika Anda RUN maka akan menghasilkan output seperti berikut:



Gambar 6.13. Output Program do-while