

Pertemuan : 12(Dua Belas)
Pokok Bahasan : Rekursif
Tujuan Khusus : Mahasiswa dapat mengerti tentang algoritma rekursif

Fungsi yang telah dibahas sebelumnya dipanggil dari bagian lain di luar tubuh fungsi yang bersangkutan. Fungsi rekursif adalah suatu fungsi yang memanggil dirinya sendiri, artinya fungsi tersebut dipanggil di dalam tubuh fungsi sendiri.

Fungsi faktorial yang menghitung nilai faktorial dari suatu bilangan bulat positif, merupakan pokok bahasan yang memudahkan pemahaman fungsi rekursif. Berikut ini adalah fungsi faktorial yang diselesaikan dengan cara biasa:

```
int faktorial (int n)
{
int counter, hasil = 1;
for (counter = n;counter > = 1, counter--)
```

hasil *=coun{

```
int counter, hasil = 1;
for (counter = n;counter > = 1, counter--)
```

hasil *=counter;

```
return hasil;
}
```

Fungsi tersebut menunjukkan bahwa nilai faktorial dihitung menggunakan looping sehingga melakukan proses sebagai berikut:

Hasil = 1;
Hasil = hasil * n; artinya hasil = n
Hasil = hasil * (n - 1);artinya hasil = n x (n - 1);

Demikian seterusnya sampai bernilai 1, atau jika dituliskan sekaligus menjadi:

Faktorial = $n! = n \times (n - 1) \times (n - 2) \dots \times 1$

Fungsi ini dapat dituliskan dalam bentuk:

Faktorial (n) = $n! = n \times (n - 1) !$

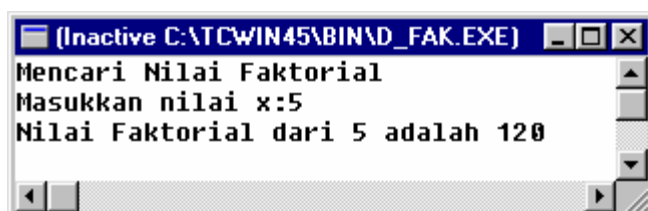
Yang menunjukkan sifat rekursif dari fungsi, yaitu $(n - 1)!$. Oleh karena itu, faktorial yang telah ditulis dalam program C++ sebelumnya, dapat ditulis kembali dalam bentuk rekursif sebagai berikut:

```
Int faktorial(int n)
{
if (n==0)
return 1;
else
return (n * faktorial (n- 1));
}
```

Program untuk kasus di atas adalah:

```
#include <iostream.h>
int faktorial(int n)
{
if (n==1)
return(1);
else
return (n*faktorial(n-1));
}
void main()
{
int x;
cout<<"Mencari Nilai Faktorial \n"<<"Masukkan nilai x:";
cin>>x;
cout<<"Nilai Faktorial dari "<<x<<" adalah
"<<faktorial(x);
}
```

output dari program di atas :



Contoh lain adalah menghitung jumlah dari suatu deret fibonacci, di mana deret tersebut didefinisikan sebagai:

Fibonacci(1) = 1

Fibonacci(2) = 1

Fibonacci(n) = fibonacci(n-1) + fibonacci(n-2)

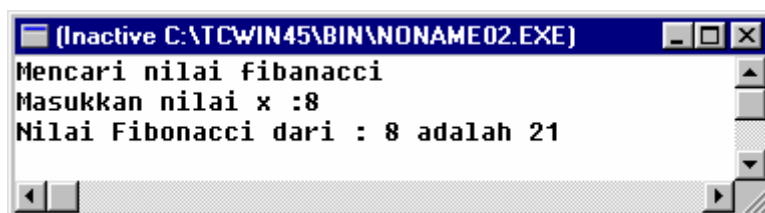
Sehingga fungsi dalam program C++ dapat dibuat sebagai berikut:

Long fibonacci (longn)

```
{  
if (n==1) || n==2)  
return 1;  
else  
return fibonacci(n-1) + fibonacci(n-2);  
}
```

Program contoh rekursi untuk fibonacci

```
#include <iostream.h>  
long fibonacci (long n)  
{  
if (( n == 1) || (n == 2))  
return 1;  
else  
return fibonacci (n-1) + fibonacci(n-2);  
}  
void main()  
{  
int x;  
cout<<"Mencari nilai fibonacci \n";  
cout<<"Masukkan nilai x :";  
cin>>x;  
cout<<"Nilai Fibonacci dari : "<<x<<" adalah  
<<fibonacci(x);  
}
```



Latihan

1. Buatlah algoritma dan program untuk menghitung pajak dari total pembelian sebuah barang, dimana pajaknya 10% dari total pembelian (Untuk menghitung pajak menggunakan fungsi)
2. Buatlah algoritma dan program dengan cara rekursi untuk menampilkan perkalian 3 buah bilangan, dimana ketiga bilangan tersebut diinputnya.
3. Buatlah algoritma dan program untuk menentukan bilangan yang dimasukkan termasuk bilangan genap atau ganjil dengan menggunakan fungsi rekursi. Syarat bilangan genap kalau bilangan itu habis dibagi nol, selainnya bilangan ganjil.

