



Modul Praktikum **Alpro Lanjut**





KONSEP LANJUTAN FUNGSI

DASAR TEORI

Pada modul I, sudah diterangkan tentang konsep dasar fungsi dan prosedur dalam C++, termasuk parameter, variabel global & local, serta cara mendeklarasikannya dan memanggil fungsi.

Pada konsep fungsi lanjutan terdapat cara membuat fungsi yang lebih fleksibel dan efisien dengan cara:

1. Fungsi Overloading
2. Keyword Return
3. Rekursif

I. FUNGSI OVERLOADING

Fungsi Overloading adalah teknik yang memungkinkan kita membuat beberapa fungsi dengan nama yang sama, tetapi dengan jumlah atau tipe parameter yang berbeda. Hal ini dapat dilakukan karena compiler akan secara otomatis memilih fungsi yang sesuai berdasarkan argumen yang diberikan saat pemanggilan.

Contoh:

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

int tambah(int a, int b)
{
    return a + b;
}

double tambah(double a, double b)
{
    return a + b;
}

int tambah(int a)
{
    return a + a;
}
```



```
int main()
{
    cout << "Hasil dari fungsi dengan tipe data int :" << tambah(5, 6) << endl;
    cout << "Hasil dari fungsi dengan tipe data double :" << tambah(5.5, 6.6) << endl;
    cout << "Hasil dari fungsi dengan tipe data int namun dengan parameter berbeda :" << tambah(5) << endl;
    return 0;
}
```

2. KEYWORD RETURN

Return dalam C++ memiliki dua fungsi utama dalam sebuah fungsi:

1. Mengembalikan nilai dari fungsi kepada pemanggilnya.
2. Menghentikan eksekusi fungsi secara langsung, meskipun masih ada kode di bawahnya.

#Contoh return mengembalikan:

return yang mengembalikan nilai disebut **fungsi**.

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

int kuadrat(int x) {
    return x * x; // Mengembalikan hasil kuadrat
}

int main() {
    int hasil = kuadrat(5);
    cout << "Hasil kuadrat: " << hasil << endl;
    return 0;
}
```

#Contoh return untuk menghentikan eksekusi:

return yang hanya menghentikan eksekusi disebut **fungsi prosedur**.



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

void cekGenap(int angka) {
    if (angka % 2 == 0) {
        cout << angka << " adalah bilangan genap." << endl;
        return; // Menghentikan eksekusi fungsi jika sudah genap
    }
    cout << angka << " adalah bilangan ganjil." << endl;
}

int main() {
    cekGenap(8); // Output: 8 adalah bilangan genap.
    cekGenap(5); // Output: 5 adalah bilangan ganjil.
    return 0;
}
```

3. REKURSIF

Rekursif adalah teknik dalam pemrograman di mana **fungsi memanggil dirinya sendiri**. Biasanya digunakan untuk masalah perulangan dan memerlukan kondisi agar berhenti.

#Contoh tanpa rekursif (Iteratif)

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

int faktorialIteratif(int n) {
    int hasil = 1;
    for (int i = 1; i <= n; i++) {
        hasil *= i;
    }
    return hasil;
}

int main() {
    cout << "5! = " << faktorialIteratif(5) << endl;
    return 0;
}
```




Base Case	Diperlukan kondisi untuk menghentikan rekursi	Tidak diperlukan
Memori	Boros (karena banyak pemanggilan fungsi)	Hemat (langsung dieksekusi)
Kecepatan	Lebih lambat karena overhead pemanggilan fungsi	Lebih cepat karena hanya menggunakan loop
Kemudahan	Lebih intuitif untuk konsep matematika	Lebih mudah dipahami dalam pemrograman
Risiko Error	Bisa menyebabkan stack overflow jika n besar	Lebih stabil

3. FILE HEADER

Pada Pemrograman C++, untuk dapat menggunakan FUNGSI `cout` dan `cin`, kita memerlukan header `iostream`, ini berarti `iostream` merupakan pustaka fungsi. Header merupakan sebuah pustaka fungsi yang dapat kita pakai. Berikut beberapa header yang dapat kita pakai.

I. `<iostream>`

Merupakan singkatan dari input output stream header yang digunakan sebagai standar input output operasi yang digunakan oleh bahasa C++. Fungsi-fungsi yang ada di dalam `iostream` antara lain sebagai berikut.

`cout` : Fungsi keluaran C++ yang menampilkan data dengan tipe data apapun ke layar

`cin` : Fungsi masukan pada c++ yang bisa memasukkan data berupa data numerik dan karakter

`endl` : Suatu Fungsi yang manipulator yang digunakan untuk melakukan perintah pindah baris



2. <conio>

Merupakan File Header yang berfungsi untuk menampilkan hasil antarmuka kepada pengguna.

Fungsi -fungsi yang ada di dalam conio antara lain sebagai berikut :

`getch()` : Singkatan dari get character and echo yang digunakan untuk menahan (pause) output suatu program dan akan kembali mengeksekusi setelah kita melakukan inputan baik itu tombol enter atau tombol lainnya dan inputan tersebut tidak ditampilkan dalam window.

`getche()` : Secara fungsi sama dengan getch() akan tetapi ketika melakukan inputan, inputan tersebut tampil dalam window.

`clrscr()` : Singkatan dari clear screen yang digunakan untuk membersihkan layar windows

3. <math.h>

Merupakan file header yang berfungsi untuk operasi matematika. Fungsi-fungsi yang ada di math.h antara lain sebagai berikut :

`sqrt()` : Fungsi ini digunakan untuk menghitung akar dari suatu bilangan. Bentuk umum penulisannya adalah sqrt(bilangan); .

`pow()` : Fungsi ini digunakan untuk menghitung hasil dari perpangkatan suatu bilangan. Bentuk umum penulisannya adalah pow(bilangan, pangkat).

`sin()`,
`cos()`,
`tan()` : Fungsi ini digunakan untuk menghitung nilai sinus, cosinus, dan tangen dari sudut yang dimasukan. Bentuk umum penulisannya adalah sin(bilangan), cos(bilangan), tan(bilangan).

4. <stdlib.h>



Merupakan file header yang berfungsi untuk operasi pembandingan dan operasi konversi.

Fungsi-fungsi yang ada di `stdlib.h` antara lain sebagai berikut :

- `max()` : Digunakan untuk menentukan nilai maksimum dari 2 bilangan yang diinputkan.
Bentuk penulisannya adalah `max(bilangan1, bilangan2);`
- `min()` : Digunakan untuk menentukan nilai minimum dari 2 bilangan yang diinputkan.
Bentuk penulisannya adalah `min(bilangan1, bilangan2);`
- `atof()` : Digunakan untuk mengkonversi nilai string menjadi bilangan bertipe double.
Bentuk umum penulisannya adalah `atof(char nama_variabel);`
- `atoi()` : Digunakan untuk mengkonversi nilai string menjadi bilangan bertipe integer.
Bentuk umum penulisannya adalah `atoi(char nama_variabel);`

5. `<string.h>`

Merupakan file header yang berfungsi untuk melakukan manipulasi string. Fungsi-fungsi yang ada di `string.h` antara lain sebagai berikut :

- `strcpy()` : Fungsi ini digunakan untuk menyalin suatu string ke variabel tujuan. Bentuk umum penulisannya adalah `strcpy(variabelTujuan, string);`
- `strlen()` : Fungsi ini digunakan untuk menghitung jumlah karakter yang ada dalam suatu string. Bentuk umum penulisannya adalah `strlen(string);`
- `strcmp()` : Fungsi ini digunakan untuk membandingkan 2 buah string. Bentuk umum penulisannya adalah `strcmp(string1, string2);` .
- `strrev()` : Fungsi ini digunakan untuk membalikan urutan suatu string. Bentuk umum penulisannya adalah `strrev(string);`
- `strlwr()` : Fungsi ini digunakan untuk mengubah semua huruf menjadi huruf kecil. Bentuk



umum penulisannya adalah `strlwr(string);`

`strupr()` : Fungsi ini digunakan untuk mengubah semua huruf menjadi huruf kapital. Bentuk umum penulisannya adalah `strupr(string);` .

`strcat()` : Fungsi ini digunakan untuk menggabungkan 2 buah string, untuk menggunakan fungsi ini juga harus menambahkan file header `ctype.h`. Bentuk umum penulisannya adalah `strcat(variabelTujuan, string);`

STUDI KASUS

Sebuah universitas di Samarinda ingin memiliki sistem sederhana untuk mencatat dan mengelola nilai mahasiswa. Program ini akan digunakan oleh bagian akademik untuk mencatat nama mahasiswa, NIM, nilai UTS, dan nilai UAS. Program ini harus memiliki beberapa fitur utama menambahkan data mahasiswa yang terdiri dari, **Nama Mahasiswa, NIM, Nilai UTS, Nilai UAS** lalu Menghitung nilai akhir dengan rumus nilai akhir= $(0.4 \times \text{UTS}) + (0.6 \times \text{UAS})$ serta menentukan status kelulusan mahasiswa berdasarkan nilai akhir:

- Lulus jika nilai akhir ≥ 60
- Tidak Lulus jika nilai akhir < 60

Menampilkan daftar mahasiswa yang sudah terdaftar beserta status kelulusannya (gunakan fungsi dan prosedur).