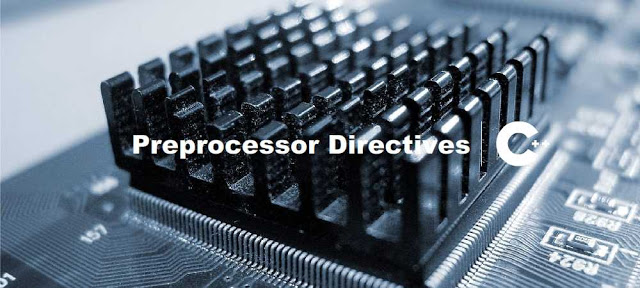
[](https://1.bp.blogspot.com/--SlVOtjWboY/VrWYYhnQRcI/AAAAAAAADEo/Ai3FzXsTDks/s1600/processor.jpg)

**Pengertian Preprocessor Directives**

Preprocessor directive merupakan suatu perintah yang termasuk kedalam program, tetapi bukanlah instruksi dari program itu sendiri, tetapi untuk preprocessor. Preprocessor ini dijalankan secara otomatis oleh kompiler, ketika didalam proses penterjemahan (Compile) program berlangsung, didalamnya membuat nilai pembuktian pertama dan menterjemahkan code program didalam kode objek. Didalam penggunaan preprocessor directive selalu dimulai dengan tanda : # Ada beberapa preprocessor directive, diantaranya adalah #define, #include, #if - #endif, # if - #else - #endif, #elif, #undef dan #ifdef - #ifndef.

**Macam-macam Preprocessor Directives**

**#define**

Digunakan untuk mendefinisikan suatu nilai tertentu kepada suatu namakonstanta. Bentuk umum dari preprocessor directive #define ini adalah:

Bentuk Penulisan :

#define nama\_konstanta teks

Contoh :

#define namanya 6

Keterangan :

* *namanya* : nama\_konstanta
* *6* : text/pernyataan

Dalam pendeklarasian preprocessor directive #define, Nama\_Konstanta sebaiknya ditulis dengan menggunakan huruf besar, guna untuk membedakannya dengan nama\_variabel. Sedangkan Teks merupakan suatu nilai yang diberikan pada nama\_konstanta. Teks dapat berupa:

* Numerik contoh: #define PI 3.14
* Karakter contoh: #define HURUF ‘B’
* String contoh: #define JABATAN “INSTRUCTOR”
* Pernyataan contoh: #define CETAK (“Borland C++”)
* Fungsi Sederhana contoh: #define LUAS\_KUBUS (n\*n)

Baca juga : [Pengertian dan Contoh Konstanta/Literal C++](http://belajarcpplus.blogspot.co.id/2016/01/pengertian-dan-contoh-konstanta.literal-c-plusplus.html)

Algoritma Pemrograman I Tim Penyusun Komisi MI 110 Setelah #define ditentukan didalam program cukup dituliskan nama\_konstantanya saja. # define akan mengganti semua nama konstanta tadi dengan teksnya sebelum proses kompilasi dimulai.

Contoh Program Penggunaan #define

#include<stdio.h>

#include<conio.h>

#include<iostream.h>

#define PI 3.141592

#define L(n) PI\*n\*n

main( )

{

   clrscr();

   cout<<"Luas Lingkaran dengan : "<<endl;

   cout<<"Jari-jari = 5 adalah "<<L(5)<<endl;

   cout<<"Jari-jari = 10 adalah "<<L(10)<<endl;

   getche( );

}

Contoh Program Penggunaan #define \*/

#include<stdio.h>

#include<conio.h>

#include<iostream.h>

#define awal {

#define akhir }

#define mulai() main()

#define cetak cout

#define masuk cin

#define hapus() clrscr()

#define tahan() getch()

#define LS\_KUBUS (sisi\*sisi)

mulai( )

awal

   int sisi, ls\_kubus;

   hapus( );

   cetak<<"Program Penggunaan #define"<<endl;

   cetak<<"Masukkan Nilai Sisi Kubus = ";

   masuk>>sisi;

   ls\_kubus = LS\_KUBUS;

   cetak<<"Luas Kubus adalah : "<<ls\_kubus;

   tahan( );

akhir

**#include**

Preprocessor #include telah dibahas pada bab sebelumnya, yaitu berfungsi untuk memasukkan atau menyertakan file-file header kedalam program yang akan dibuat. Dalam penulisan #include ada dua bentuk penulisan :

#include "nama\_file\_header"

atau

#include <nama\_file\_header>

Pada bentuk penulisan #include mempunyai arti yang berbeda, yaitu:

•   #include "nama\_file\_header"

“Pertama kali compiler akan mencari file header yang disebutkan pada directori yang sedang aktif dan apa bila tidak ditemukan akan mencari pada directori dimana file header tersebut berada “.

•   #include <nama\_file\_header>

“Pertama kali compiler akan mencari file header yang disebutkan pada directori yang ada file headernya, kecuali pada directori yang sedang aktif.

Baca juga : [Pengertian dan Cara Pembuatan File Header C++](http://belajarcpplus.blogspot.com/2016/01/pengertian-dan-cara-pembuatan-file-header-cplusplus.html)

**#if - #endif**

Preprocessor #if - #endif digunakan untuk mengkompilasi jika pernyataan kondisi pada #if bernilai benar, jika tidak maka, diabaikan. Pernyataan kondisi berupa ekspresi konstanta yang dideklarasikan dengan #define.

Benuk Penulisan

#if ekspresi-konstanta

penyataan;

#endif

Contoh Penggunaan #if - #endif

#include <conio.h>

#include <stdio.h>

#define N 4

main( )

{

   #if N > 0

   printf("Lebih Besar dari Nol");

   #endif

   getch( );

}

Baca juga : [Pengertian dan Contoh Pernyataan IF C++](http://belajarcpplus.blogspot.com/2016/01/pengertian-dan-contoh-pernyataan-if-cplusplus.html)

**# if - #else - #endif**

Preprocessor #if - #else -#endif digunakan untuk mengkompilasi jika pernyataan kondisi pada #if bernilai benar, jika #if bernilai salah maka, pernyataan #else dikompilasi. Pernyataan kondisi berupa ekspresi konstanta yang dideklarasikan dengan #define.

Benuk Penulisan

#if ekspresi-konstanta

   Penyataan-1;

#else

   Penyataan-2;

#endif

Contoh Penggunaan #if - #else - #endif

#define N -4

main( )

{

   #if N > 0

      printf("Lebih Besar dari Nol");

   #else

      printf("Lebih Kecil dari Nol");

   #endif

}

Baca juga : [Macam-macam dan Contoh Manipulator C++](http://belajarcpplus.blogspot.com/2016/01/macam-macam-dan-contoh-manipulator-cplusplus.html)

**# elif**

Preprocessor #elif digunakan untuk mengkompilasi dari pernyataan bertingkat. Dalam hal ini #elif sama halnya seperti #elseif, merupakan kombinasi dari #if dan #else. Perintah dijalankan sesuai dengan kondisi yang telah ditentukan, Hasil hanya dapat dijalankan sesuai dengan ketentuan yang benar. Bentuk #elif diikuti oleh ekspresi-konstanta.

Benuk Penulisan

#if ekspresi-konstanta-1

   Penyataan-1;

#elif ekspresi-konstanta-2

   Penyataan-2;

# elif ekspresi-konstanta-n

   Penyataan-n;

#endif

Contoh Penggunaan #elif

#define N 12

main( )

{

   #if N > 10

      printf("Lebih Besar dari Sepuluh");

   #elif N == 10

      printf("Sama Dengan Sepuluh ");

   #else N < 10

      printf("Lebih Kecil dari Sepuluh");

   #endif

}

Baca juga : [Pengertian Prototipe dan Parameter Function C++](http://belajarcpplus.blogspot.com/2016/01/pengertian-prototipe-dan-parameter-function-cplusplus.html)

**#undef**

Preprocessor #undef digunakan untuk menghilangkan nilai yang telah didefiniskan dari daftar definisi.

Contoh Penggunaan #undef

#include<iostream.h>

#define LEBAR\_MAKS 100

#if LEBAR\_MAKS>200

#undef LEBAR\_MAKS //--> menghilangkan LEBAR\_MAKS

#define LEBAR\_MAKS 200

#elsif LEBAR\_MAKS <50

#undef LEBAR\_MAKS //--> menghilangkan LEBAR\_MAKS

#define LEBAR\_MAKS 50

#else

#undef LEBAR\_MAKS //--> menghilangkan LEBAR\_MAKS

#define LEBAR\_MAKS 50

#endif

main( )

{

   char str[LEBAR\_MAKS];

   cout<<LEBAR\_MAKS;

}

Baca juga : [Pengertian dan Keuntungan Object Oriented Programming (OOP) C++](http://belajarcpplus.blogspot.co.id/2016/01/pengertian-dan-keuntungan-object-oriented-programming-oop-cplusplus.html)

**# ifdef - # ifndef**

Preprocessor #ifdef dan #ifendef memberikan bagian dari program yang akan dikompile, hal ini dapat dilakukan jika sudah konstanta didefiniskan pada bagian #define, hal ini merupakan parameter yang khusus yang harus terdefinisi.

Benuk umum penulisan :

#ifdef

nama-konstanta pernyataan;

#endif

Penjelasan: Jika nama-konstanta terdefinisi maka, pernyataan akan dijalankan, jika nama-konstanta tidak terdefinisi maka, pernyataan akan diabaikan.

Contoh  Penggunaan #ifdef dan #ifndef

#include<iostream.h>

#define ANAK1 "ILHAM"

#define ANAK2 "HADIANSYAH"

main( )

{

   #ifdef ANAK1

      cout<<"Pagi ini "<<ANAK1<<" pergi Kesekolah";

   #else

   #ifndef ANAK2

      cout<<"Hari ini "<<ANAK1<<" "<<ANAK2;

      cout<<" pergi Kesekolah";

   #else

      cout<<"Pagi ini "<<ANAK2<<"pergi Kesekolah";

   #endif

   #endif

   getche();

}

Baca juga : [Pengertian Input dan Output C++](http://belajarcpplus.blogspot.co.id/2016/01/pengertian-input-dan-output-c-plus-plus.html)

Cukup sekian dari belajarcpp.com semoga artikel ini bisa bermanfaat untuk kita semua. Mohon maaf atas kekuranganya dan Terima Kasih atas dukungan dan kunjungan anda ke BelajarCPP. Have a nice day.

Last update : 6 Februari 2016