* **Fungsi #include** adalah sebuah prosesor pengarah yang mengatakan kepada kompiler untuk meletakan kode dari header file iostream.h kedalam program. Fungsi cout memerlukan file iostream.h.
* **Fungsi Main** adalah nama judul fungsi.
* Fungsi Tanda **()**digunakan untuk mengapit argumen fungsi yaitu nilai yang akan dilewatkan ke fungsi.
* Fungsi Tanda **{**pada fungsi main() menyatakan awal eksekusi program. Adapun } pada fungsi main() menyatakan akhir eksekusi program.
* Fungsi **cout** (character out) dipakai untuk menampilkan text di layar monitor anda. Cout merupakan sebuah objek di dalam C++, yang digunakan untuk mengarahkan data ke dalam standar output (cetak pada layar)
* Fungsi Tanda **<<**, yang diketahui sebagai operator pemasukan (insertion operators). Tanda tersebut mengatakan kepada kompiler agar segera menghasilkan output sesuai dengan input anda.
* **”Mugi Rachmat“** merupakan sebuah pernyataan yang digunakan untuk menampilkan tulisan yang terletak pada sepasang tanda petik ganda ke layar. Tulisan yang terletak pada sepasang tanda petik ganda disebut string. Setiap pernyataan harus diakhiri dengan tanda titik koma (;).
* Fungsi **Tanda semicolon “ ; ”**digunakan untuk mengakhiri sebuah pernyataan. Setiap pernyataan harus diakhiri dengan sebuah tanda semicolon.
  1. **include**  
       
     Adalah salah satu pengarah prepocessor directive yang tersedia pada C++.  
     Preprocessor selalu dijalankan terlebih dahulu pada saat proses kompilasi  
     terjadi. Bentuk umumnya :  
       
     **# include <nama\_file>**  
       
     tidak diakhiri dengan tanda semicolon, karena bentuk tersebut bukanlah suatu  
     bentuk pernyataan, tetapi merupakan prepocessor directive. Baris tersebut  
     menginstrusikan kepada kompiler untuk menyisipkan file lain dalam hal ini file  
     yang berakhiran .h (file header) yaitu file yang berisi C++ standard library.  
     contohnya:  
       
     - **#include <iostream.h&g**t; : diperlukan pada program yang melibatkan objek **cout** dan **cin**  
     -**#include <conio.h>** : diperlukan bila melibatkan **clrscr()**, yaitu perintah untuk membersihkan layar dan fungsi **getch()**untuk menerima sembarang input keyboard dari user.  
     - **#include <iomanip.h>**: diperlukan bila melibatkan **setw()**yang bermanfaat untuk mengatur lebar dari suatu tampilan data.  
     - **#include <math.h>** : diperlukan pada program yang menggunkan operasi **sqrt()**yang bermanfaat untuk operasi matematika kuadrat.  
       
     **2.Fungsi main ()**  
       
     Program C++ terdiri dari satu atau lebih fungsi, dan di antara salah  
     satunya harus ada fungsi main dan hanya boleh ada satu main pada tiap  
     program C++. Setiap program C++ akan dan pasti akan memulai eksekusi  
     programnya pada fungsi main ini, meskipun main bukan fungsi yang pertama  
     ditulis di program.  
       
     Melihat bentuk seperti itu dapat kita ambil kesimpulan bahwa batang  
     tubuh program utama berada didalam fungsi main(). Berarti dalam setiap  
     pembuatan program utama, maka dapat dipastikan seorang pemrogram  
     menggunakan minimal sebuah fungsi.  
       
     Tanda { dan pada akhir program terdapat tanda }. Tanda { harus ada pada  
     setiap awal dari sebuah fungsi dan tentu saja harus diakhiri dengan tanda }.  
     Tanda ini digunakan untuk menunjukkan cakupan(scope) dari sebuah fungsi,  
     dimana untuk menunjukkan fungsi ini dimulai dan berakhir.  
       
     **3. Komentar**  
       
     Komentar tidak pernah dicompile oleh compiler. Dalam C++ terdapat 2  
     jenis komentar, yaitu:  
       
     **Jenis 1 : /\* Komentar anda diletakkan di dalam ini  
     Bisa mengapit lebih dari satu baris \*/  
     Jenis 2 : // Komentar anda diletakkan disini ( hanya bisa sebaris )**  
       
     Programmer sering sekali memasukkan komentar di dalam code agar program  
     lebih mudah dibaca. Komentar juga membantu orang lain untuk membaca dan  
     mengerti isi dari code. Komentar tidak menyebabkan komputer melakukan suatu  
     instruksi ketika program dijalankan.  
       
     **4.Tanda Semicolon**  
     Tanda semicolon **“ ; ”** digunakan untuk mengakhiri sebuah pernyataan. Setiap pernyataan harus diakhiri dengan sebuah tanda semicolon.  
       
     **5. Mengenal Input/Output**  
     Pernyataan cout (dibaca C out) merupakan sebuah objek di dalam C++,  
     yang digunakan untuk mengarahkan data ke dalam standar output (cetak pada  
     layar). Sedangkan untuk menginputkan data, dapat digunakan cin (dibaca C in).  
       
     Berikutnya adalah operator **<<** Operator ini digunakan sebagai penghubung antara stream dengan kalimat. Operator ini disesuaikan denganfungsional dari cout.   
     Untuk sementara bayangkan saja operator **<<**sebagai arah dari aliran data. Jadi karena kita ingin mencetak kalimat ke layar, dan yang menghubungkan program kita dengan layar dengan cout, otomatis kita harus mengirimkan kalimat ke cout. Maka operator **<<**digunakan, yang berarti kalimat dialirkan ke arah cout, dan cout akan mencetaknya ke layar.  
       
     Sintaks yang digunakan :  
       
      **cout << daftar\_keluaran  
     cin >> daftar\_masukan**  
       
     **endl** merupakan suatu fungsi manipulator yang digunakan untuk menyisipkan karakter NewLine atau mengatur pindah baris. Fungsi ini sangat berguna untuk piranti keluaran berupa file di disk. File header yang harus disertakan adalah file **header iostream.h**   
       
     **Fungsi getch()**(get character and echo) dipakai untuk membaca sebuah karakter dengan sifat karakter yang dimasukkan tidak perlu diakhiri dengan menekan tombol ENTER, dan karakter yang dimasukan tidak akan ditampilkan  
     di layar. File header yang harus disertakan adalah **conio.h**
* ***Char berfungsi tipe data karakter dan memanggil nya dengan %s***
* ***Int berfungsi sebagai tipe data angka genap asli dan di panggil menggunakan %d***
* ***Float berfungsi sebagai tipe data angka yg berkoma dan di panggil menggunakan %f***

 kita mulai dari printf. Printf merupakan sebuah fungsi dalam file header <stdio. h>. Printf digunakan untuk membuat sebuah output berupa tampilan. contoh:

*#include <stdio.h>*

*void main()*

*{*

*printf (” ILC DONK “) ;*

*}*

kata yang akan di tampilkan di ketik di antara tanda (” “). dan dalam setiap statement harus di akhiri dengan tanda” ; “.

selanjutnya kita bahas tentang scanf.

scanf merupakan sebuah fungsi dalam file header <stdio.h> juga yang berfungsi untuk menerima inputan dari user. Untuk belajar tentang scanf kita pelajari dulu tentang variabel. Variabel di ibaratkan sebuah wadah untuk menampung sebuah nilai maupun karakter dari inputan user ataupun sudah di tentukan dari awal. Ada banyak macam tipe variabel

* int -> untuk menampung nilai
* char -> untuk menampung karakter
* dll
* yang pernah saya tulis pada  [memahami fungsi if](http://pemrograman-c.blogspot.com/2011/11/memahami-fungsi-peryataan-if-dalam.html) adalah untuk mengeksekusi program bila kondisi bernilai benar. Proses eksekusi kemudian berlanjut pada pernyataan berikutnya secara urut. C++ juga memiliki versi pernyataan if yang memungkinkan untuk mengeksekusi suatu pernyataan jika kondisi bernilai benar dan akan mengeksekusi pernyataan lainnya jika kondisi bernilai salah. pengalihan eksekusi dapat dilakukan dengan menggunakan kata kunci **else**. Proses eksekusi kemudian berlanjut pada pernyataan berikutnya yang berada diluar kotak blog pernyataan if.

| **Format Data** | **Penjelasan** | **Contoh** |
| --- | --- | --- |
| **%c** | Karakter tunggal | a, b, 1 |
| **%d** | Bilangan bulat *(Integer)* | -2, 31, 45 |
| **%e** | Notasi ilmiah *(Lowercase)* | 3.141600e+000 |
| **%E** | Notasi ilmiah *(Uppercase)* | 3.141600E+000 |
| **%f** | Bilangan pecahan bertitik *(Floating Number)* | 3.14 |
| **%g \*** | Menggunakan tampilan terpendek dari %e dan %f | 392.65 |
| **%G \*** | Menggunakan tampilan terpendek dari %E dan %F | 392.65 |
| **%i** | Bilangan integer | -1, 2, 100 |
| **%n \*** | Menyimpan jumlah karakter yang telah dituliskan |  |
| **%o** | Bilangan oktal | 100 => 144 |
| **%p \*** | Pointer |  |
| **%s** | String atau deret karakter | Ayub, belajar C++ |
| **%u \*** | Bilangan bulat desimal tidak bertanda*(Unsigned)* | 7235 |
| **%x** | Bilangan bulat hexadecimal *(Lowercase)* | 100 => 64 |
| **%X** | Bilangan bulat hexadecimal *(Uppercase)* | 100 => 64 |
| **%%** | Tanda persen | 50%, %, -20% |

|  |  |  |
| --- | --- | --- |
| **TIPE** | **DEFINISI** | **NILAI** |
| **int** | Bilangan Integer | -32768 to 32767 |
| **float** | Bilangan Real | +/- 3.4e +/- 38 (~7 digits) |
| **double** | Bilangan Real dengan rentang lebih luas | +/- 1.7e +/- 308 (~15 digits) |
| **long double** | Bilangan Real dengan rentang paling luas | +/- 3.4e +/- 308 (~15 digits) |
| **char** | Karakter : bisa alfabet, simbol, operator, karakter khusus | Misal : A, b, $, + dan lainnya |
| **boolean** | Kondisi benar atau salah | Hanya ada nilai **true** untuk benar dan**false** untuk salah |

stdlib.h

File header stdlib.h menjabarkan beberapa fungsi umum dan marco termasuk manajemen memori dinamis, menjalin komunikasi dengan perangkat sekitar, membuat bilangan secara random, aritmatika bilangan integer, pencarian, pengurutan dan pengkonversian.

iostream

File header ini ditulis dengan diikuti using namespace std;. Memiliki beberapa perintah umum seperti cout untuk output dan cin untuk input.

stdio.h

File header ini merupakan singkatan dari standar input output. Jadi sudah jelas kalau perintah yang ada dijadikan sebagai standar input dan output. Contoh perintah : scanf, gets, printf, puts, putchar.

conio.h

File header ini merupakan singkatan dari Console Input and Output dan digunakan untuk membuat teks antarmuka pengguna. Contoh perintah : getchar, getch, getche, clrscr().

math.h

File header ini mengandung beberapa fungsi yang berhubungan dengan matematika. Contoh fungsi : sqrt(), po(), log10(), sin().

string.h

File header ini digunakan untuk memanipulasi sting dan array. Contoh fungsi : strcpy(), strlen(), strcat().

errno.h

File header ini berfungsi untuk melaporkan kesalahan yang terjadi melalui suatu kode kesalahan (*error number*).

ctype.h

File header ini mendeklarasikan seperangkat fungsi untuk mengklasifikasi dan mentransformasi karakter.

time.h

File header ini mengandung beberapa definisi fungsi untuk memanipulasi informasi tanggal dan waktu.

Berikut ini adalah bebarapa hal yang perlu diketahui tentang Variabel :

1. Pemberian nama variabel harus **unik**, tidak boleh sama dengan kata kunci bawaan C++ maupun fungsi.

2.  Dalam nama sebuah variabel **tidak boleh mengandung spasi**.

3.  Nama variabel bersifat ***case sensitive***.

4.  Nama variabel hanya terdiri dari **huruf, angka dan tanda *undescore* “\_”**. Tidak boleh ada karakter atau simbol khusus.

5. Nama variabel hanya boleh **diawali dengan huruf dan *underscore***, tidak boleh diawali angka atau karakter/simbol khusus.

6.  Jika nama sebuah variabel lebih dari 32 karakter, maka yang akan terbaca komputer hanya**32 karakter pertama** saja, selebihnya tidak akan dianggap tetapi masih dapat dieksekusi.

**Pendeklarasian Variabel**

**2. puts()**  
Perintah puts() sebenarnya sama dengan printf(), yaitu digunakan untuk mencetak string ke layar. puts() berasal dari kata PUT STRING.   
Perbedaan antara printf() dengan puts() adalah :  
printf() puts()  
Harus menentukan tipe data untuk data tring, yaitu %s Tidak Perlu penentu tipe data string, karena fungsi ini khusus untuk tipe data string.  
Untuk mencetak pindah baris, memerlukan notasi ‘ \n ‘ Untuk mencetak pindah baris tidak perlu notasi ‘ \n ‘, karena sudah dibeikan secara otomatis.

**Penggunaan escape Sequence**

Penggunaan escape  sequence menngunakan notasi  “\” (back slash ) jika karakter terdapat notasi ini sebagai karakter “escape” (menghindar)

berikut adalah escape sequence characters dalam C++ tersebut:

: karakter nol atau karakter ber-ASCII nol.

\a : untuk membunyikan sound beep

\b : untuk backspace \f : untuk ganti halaman (formfeed)

\n : untuk membuat barus baru (newline) fungsi sama seperti endl

\r : untuk kembali ke awal baris atau carriage return

\t : untuk tab horizontal, fungsi sama ketika kita mengetik di word, lalu menekan tombol tab di keyboard

\v : untuk membuat tab vertikal

\\ : untuk membuat karakter

\ \’ : untuk membuat karakter

‘ \” : untuk membuat karakter

” \? : untuk membuat karakter ?

\ooo : untuk karakter yang nilai oktalnya sebanyak tiga digit oktal

\xhh : untuk karakter yang nilai heksadesimalnya sebanyak dua digit heksadesimal

**Puts ()**

**do**ini merupakan statement dari do while yaitu untuk kembali ke program menu, jadi jika memilih program dari ke 3 pilihan tersebut maka program akan kembali ke menu awal.

**switch(pil)**switch ini digunakan untuk membandingkan beberapa nilai konstan yang mungkin untuk sebuah ekspresi, hampir sama dengan if dan else if.

**{**

Untuk akhir dari perintah dalam case digunakan **break;** .

Perintah default digunakan ketika statement yang diperintah di luar dari perintah case-case sebelumnya.

a. Pernyataan Do…While

Digunakan untuk menjalankan suatu pernyataan lalu diuji apakah kondisi sudah bernilai benar, tapi jika bernilai salah maka pernyataan tidak dijalankan. Pada dasarnya perulangan do….while sama saja dengan perulangan while, hanya saja pada proses perulangan dengan while, seleksi berada di while yang letaknya di atas sementara pada perulangan do….while, seleksi while berada di bawah batas perulangan. Jadi dengan menggunakan struktur do…while sekurang-kurangnya akan terjadi satu kali perulangan.

Contoh Program :

b. Pernyataan While

Perulangan while banyak digunakan pada program yang terstruktur. Perulangan ini banyak digunakan bila jumlah perulangannya belum diketahui. Proses perulangan akan terus berlanjut selama kondisinya bernilai benar (true) dan akan berhenti bila kondisinya bernilai salah atau batas yang ditentukan telah terpenuhi..

Contoh Program :