Pengertian dan Fungsi Boolean

Bukan hanya C++ yang menggunakan Operasi Boolean / aljabar Boolean. Boolean juga ada pada beberapa bahasa pemrograman lainya seperti C, C#, Javascript, PHP dan lain-lain. Boolean juga ada di dalam lmu matematika. Pengertian Boolean dalam ilmu Matematikan dan Komputer adalah Struktur aljabar yang mencangkup intisari dari operasi logika AND, OR, XOR (Exclusive Or) dan NOT.

Boolean adalah suatu tipe data yang hanya mempunyai dua nilai yaitu True(Benar) atau False(Salah). Boolean mempunyai nama Operasi Boolean yang berasal dari nama seorang matematikawan asal inggris, bernama George Boole (1855-1864). Merupakan seseorang yang pertama kali mengenalkan istilah Boolean di pertengahan abad ke-19.

Semua operasi AND, OR, XOR (Exclusive Or) dan NOT memiliki sifat sendiri-sendiri yang membuat mereka pada karakteristik dalam pembacaan operasi Boolean.

Macam-Macam Operasi Boolean

AND ( & )

Operasi ini dilakukan antara dua bit/ekspresi, menghubungkan dan membandingkan mereka sehingga menghasilkan satu pilihan dari dua pilihan. Yaitu BENAR/TRUE/1 atau SALAH/FALSE/0. Operasi ini akan menghasilkan BENAR/TRUE/1 jika dari kedua atau semua bit/ekspresi bernilai BENAR/TRUE/1. Dan akan menghasilkan nilai SALAH/FALSE/0 jika dari semua perbandingan memiliki nilai SALAH/FALSE/0.

|  |  |  |
| --- | --- | --- |
| X | Y | X & Y |
| 1 | 1 | 1 |
| 1 | 0 | 0 |
| 0 | 1 | 0 |
| 0 | 0 | 0 |

OR ( | )

Operasi ini dilakukan antara dua bit/ekspresi, menghubungkan dan membandingkan mereka sehingga menghasilkan satu pilihan dari dua pilihan. Yaitu BENAR/TRUE/1 atau SALAH/FALSE/0. Operasi ini akan menghasilkan BENAR/TRUE/1 jika dari kedua atau semua bit/ekspresi mempunyai nilai BENAR/TRUE/1. Dan akan menghasilkan nilai SALAH/FALSE/0 jika dari semua bit/ekspresi bernilai SALAH/FALSE/0.

|  |  |  |
| --- | --- | --- |
| X | Y | X | Y |
| 1 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

XOR ( ^ )

Xor ( ^ ) digunakan untuk membandingkan bit/ekspresi. Akan bernilai benar (1) jika dari dua bit yang dibandingkan akan dinyatakan BENAR/TRUE/1 jika ada sebuah bit/ekspresi bernilai BENAR/TRUE/1 dan SALAH/FALSE/0 atau dengan kata lain dari semua bit/ekspresi memiliki semua nilai dari 2 pilihan yaitu BENAR/TRUE/1 dan SALAH/FALSE/0. Dan Akan dinyatakan SALAH/FALSE/0 jika semua bit/ekspresi bernilai SALAH atau BENAR.

|  |  |  |
| --- | --- | --- |
| X | Y | X ^ Y |
| 1 | 1 | 1 |
| 1 | 0 | 1 |
| 0 | 1 | 1 |
| 0 | 0 | 0 |

NOT ( ~ )/( ! )

Fungsi NOT ( ~ )/( ! ) adalah untuk membalikan nilai dari bit/ekspresi. NOT Akan Menyatakan BENAR/TRUE/1 pada hasil yang bernilai SALAH/FALSE/0, dan Akan dinyatakan SALAH/FALSE/0 pada hasil yang bernilai BENAR/TRUE/1.

Perbedaan [!] dan [~]

[!] Digunakan untuk mengoperasikan ekspresi, Biasanya digunakan pada Operasi Boolean dan Logika

[~] Digunakan untuk mengoperasikan bilangan pada level Biner, Biasanya digunakan dalam Operasi Bitwise

|  |  |
| --- | --- |
| Nilai | Hasil |
| 1 | 0 |
| 0 | 1 |

Contoh Program Operasi Boolean :

#include <iostream>

using namespace std;

int main( ){

bool hasilBool;

int x=0,y=0,hasilInt=0;

cout<<"Masukkan Nilai X = ";cin>>x;

cout<<"Masukkan Nilai Y = ";cin>>y;

//DAN (&)

hasilBool=x>9&y<9;

cout<<"AND (&) -> "<<(x>9)<<" & "<<(y<9)<<" = "<<hasilBool<<endl;

//OR (|)

hasilBool=x>9|y>9;

cout<<"OR (|) -> "<<(x>9)<<" | "<<(y>9)<<" = "<<hasilBool<<endl;

//XOR (^)

hasilBool=x>9^y<=10;

cout<<"XOR (^) -> "<<(x>9)<<" ^ "<<(y<=10)<<" = "<<hasilBool<<endl;

//NOT (!)/(~)

hasilBool=!(x>=10);

cout<<"NOT (!) -> "<<(x>=10)<<" = "<<hasilBool<<endl;

hasilInt=~x;

cout<<"NOT (~) -> "<<x<<" = "<<hasilInt<<endl;

return 0;

}

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.