Pengertian dan Contoh Compound Assignment

Pengertian Pemberi Nilai Aritmetika

“Compound Assignment” jika diterjemahkan ke dalam bahasa indonsia adalah “Senyawa tugas”, tetapi di Indonesia kebanyakan orang menyebutnya dengan nama “Pemberi Nilai Aritmetika”. Operator ini berfungsi untuk memodifikasi nilai dari variabel yang dituju dengan sumber nilai langsung, nilai dari variabel lain atau nilai dari variabel itu sendiri sendiri. Compound Assigment memiliki bentuk-bentuk operator, yaitu :

|  |  |  |
| --- | --- | --- |
| Operator | Keterangan | Contoh |
| += | Penambahan | X += Y |
| -= | Pengurangan | X -= Y |
| \*= | Perkalian | X \*= Y |
| /= | Pembagian | X /= Y |
| %= | Sisa Pembagian (Modulo) | X %= Y |
| >>= | Pergeseran bit ke kiri | X >>= Y |
| <<= | Pergeseran bit ke kanan | X <<= Y |
| &= | Bitwise DAN | X &= Y |
| ^= | Bitwise OR Eksklusif | X ^= Y |
| |= | Bitwise OR Inklusif | X |= Y |

Tanda-tanda dari Operator Compound Assignment di atas semuanya hampir mirip seperti tanda pada beberapa operator lainya di artikel-artikel sebelumnya, hanya berbeda pada penambahan tanda “=” setelah tanda inti. Untuk masing-masing fungsinya sebenarnya masih sama seperti apa yang sudah dijelaskan pada artikel sebelumnya di sini hanya berbeda pada penulisanya yang lebih singkat, hal ini dapat menyingkat waktu dan dapat mempermudah programmer untuk melakukan operasi pada suatu data. Dibawah dijabarkan arti dari penulisan beberapa operator di atas.

|  |  |  |
| --- | --- | --- |
| No | Penyingkatan | Arti |
| 1 | X += Y | X = X + Y |
| 2 | X -= Y | X = X – Y |
| 3 | X \*= Y | X = X \* Y |
| 4 | X /= Y | X = X / Y |

Contoh Penulisan

X += 2 // x = x + 2

//penulisan di bawah juga berlaku

X \*= Y \* 3 / x = x \* (y + 3)

Contoh Program :

#include <iostream>

using namespace std;

int main ()

{

int x = 10, y = 2;

cout<<(x += y)<<endl;

x=10;

cout<<(x -= y)<<endl;

x=10;

cout<<(x \*= y)<<endl;

x=10;

cout<<(x /= y)<<endl;

x=10;

cout<<(x %= y)<<endl;

x=10;

cout<<(x >>= y)<<endl;

x=10;

cout<<(x <<= y)<<endl;

x=10;

cout<<(x &= y)<<endl;

x=10;

cout<<(x ^= y)<<endl;

x=10;

cout<<(x |= y)<<endl;

return 0;

}