

Operator merupakan simbol yang biasa dilibatkan dalam program untuk melakukan suatu operasi atau manipulasi, sedangkan ungkapan adalah dasar bagi pernyataan berkondisi (misalnya if). Hasil ungkapan berupa 1 kalau ungkapan bernilai benar dan ungkapan berupa 0 kalau ungkapan bernilai salah.

Operator merupakan simbol yang biasa dilibatkan dalam program untuk melakukan suatu operasi atau manipulasi. Sebagian operator C++ tergolong sebagai operator binary, yaitu operator yang dikenakan terhadap dua buah nilai (operand). simbol “+” merupakan operand untuk melakukan penjumlahan dari a dan b. Karena operator penjumlahan melibatkan dua operand, operator penjumlahan tergolong sebagai operator binary. simbol “-” (minus) merupakan unary, karena hanya memiliki sebuah operand (yaitu c pada contoh diatas). Ungkapan (ekspresi) dalam C++ dapat berupa: Pengenal, Konstanta, Diantara kombinasi elemen diatas dengan operator.

Operator untuk aritmatika yang tergolong sebagai operator binary. Contoh penggunaan operator aritmatika misalnya untuk memperoleh nilai diskriminan dari suatu persamaan kuadrat. Operator aritmatika mempunyai prioritas pengerjaan. Prioritas yang tinggi akan diutamakan dalam hal pengerjaan dibandingkan dengan operator yang memiliki prioritas yang lebih rendah. Jika operator memiliki prioritas yang sama, operator sebelah kiri akan diutamakan untuk dikerjakan terlebih dahulu. Tanda kurung biasa digunakan untuk merubah urutan pengerjaan. Operator sisa pembagian (operator modulus) yang berupa %. Operator ini diterapkan pada operand bertipe integer. Kegunaan operator % diantaranya bisa dipakai untuk menentukan suatu bilangan bulat termasuk ganjil atau genap.

Kedua operator ini digunakan pada operand bertipe bilangan bulat. Operator penaikan digunakan untuk menaikkan nilai variable sebesar satu, sedangkan operator penurunan dipakai untuk menurunkan nilai variabel sebesar satu.

Operator kondisi biasa dipakai untuk mendapatkan sebuah nilai dari dua buah kemungkinan, berdasarkan suatu kondisi.

Ungkapan adalah ungkapan yang menjadi dasar bagi pernyataan berkondisi (misalnya if). Hasil ungkapan berupa 1 kalau ungkapan bernilai benar dan ungkapan berupa 0 kalau ungkapan bernilai salah. Operator biasa digunakan untuk membandingkan dua buah nilai. Agar tidak salah dalam menuliskan suatu ungkapan,

pengetahuan tentang prioritas operator perlu diketahui. pada pernyataan diatas, operator yang dilibatkan (=) mempunyai sifat pengerjaan dimulai dari kanan.

Dalam melakukan operasi seperti memperoleh akar kuadrat ataupun memperoleh logaritma alamiah dari suatu nilai. Pada C++ memang tidak terdapat operator-operator yang khusus untuk melaksanakan operasi-operasi seperti itu. Tetapi tidak berarti C++ tidak dapat melakukan operasi itu. C++ menyediakan sejumlah fungsi pustaka (library functions) yang dirancang untuk memenuhi solusi dari berbagai persoalan.

Misalkan kita akan menghitung sebuah akar kuadrat, pemrogram bisa menggunakan fungsi `sqrt()`. Jika program ingin menggunakan fungsi pustaka, perlulah untuk mencatumkan deklarasi dari fungsi bersangkutan. Untuk keperluan ini program mesti menyertakan baris: `#include <nama_file>` dengan `nama_file` adalah nama header, yaitu file yang berakhiran `.h`. sebagai contoh program diatas menyertakan `#include <math.h>` disebabkan file header tersebut berisi deklarasi (prototipe) dari fungsi `sqrt()`.

Tipe data dasar adalah tipe data yang dapat langsung digunakan, dan memiliki ukuran tertentu sesuai dengan tipe data masing-masing. Beberapa tipe data dasar: integer, real, char, string, dan boolean. Variabel adalah obyek yang nilainya dapat berubah-ubah dalam sebuah program, sedangkan konstanta memiliki nilai yang tetap sejak dideklarasikan hingga program berakhir. Pada saat mendeklarasikan sebuah variabel, pemrogram harus menyebutkan nama variabel dan tipe data dari variabel tersebut. Operator adalah pengendali operasi yang akan dilakukan pada beberapa operand sehingga membentuk sebuah ekspresi. Tiga macam operator dalam pemrograman: operator aritmatik, operator relasional, dan operator logika. Penggunaan beberapa operator sekaligus dalam sebuah ekspresi mengakibatkan perlunya menentukan urutan operasi yang diatur dalam level urutan operator. Operator dengan level yang lebih tinggi akan dieksekusi lebih dahulu daripada operand dengan level yang lebih rendah. Operator-operator yang memiliki level yang sama akan dieksekusi menurut urutan penulisan (dari kiri ke kanan).