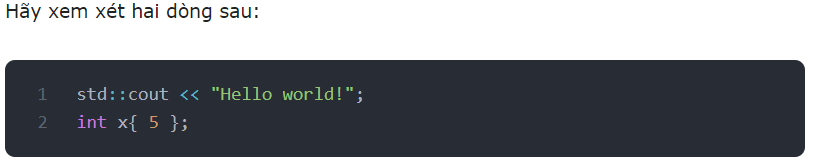
**GIỚI THIỆU LITERALS VÀ OPERATORS**

**1. Literals**



“Hello world” là gì? Nó là literal, một chữ (literal) còn được gọi là hằng số kiểu chuỗi (literal constant) là một giá trị cố định đã được chèn trực tiếp vào code.

Tuy nhiên, giá trị của một chữ (Literals) là cố định và không thể thay đổi (do đó nó được gọi là hằng số), trong khi giá trị của một biến có thể được thay đổi thông qua khởi tạo và gán.

**2. Operator (Toán tử)**

Trong toán học, một phép toán là một phép tính liên quan đến 0 hoặc nhiều giá trị đầu vào (được gọi là toán hạng) tạo ra một giá trị mới (được gọi là giá trị đầu ra). Một hoạt động cụ thể được thực hiện bằng một cấu trúc nào đó (thường là ký hiệu hoặc cặp ký hiệu) được gọi là toán tử.

Ví dụ, như trẻ em, tất cả chúng ta đều học rằng 2 + 3 bằng 5. Trong trường hợp này, 2 và 3 là các toán hạng và ký hiệu + là toán tử cho chúng ta áp dụng phép toán cộng trên toán hạng để tạo ra giá trị mới 5.

Ví dụ: toán tử cộng sẽ được gọi là toán tử + và toán tử trích xuất sẽ được gọi là toán tử >>.

Bạn có thể đã khá quen thuộc với các toán tử số học từ việc sử dụng khá phổ biến trong toán học, bao gồm phép cộng (+), phép trừ (-), phép nhân (\*) và phép chia (/). Trong C++, gán (=) cũng là một toán tử, cũng như << (chèn) và >> (trích xuất). Một số toán tử được sử dụng nhiều hơn một ký hiệu, chẳng hạn như toán tử đẳng thức (==), cho phép chúng ta só sánh hai giá trị để xem chúng có bằng nhau hay không. **Ngoài ra còn có một số toán tử là các từ (ví dụ: new, delete và thrown).**

**Các toán tử trong C++ có ba loại khác nhau:**

**Toán tử đơn (Unary operators) chỉ hành động trên một toán hạng**. Một ví dụ về toán tử đơn là toán tử -. Ví dụ, đã cho -5, toán tử - lấy toán hạng bằng 5 và lật dấu của nó để tạo ra giá trị đầu ra mới -5.

**Toán tử nhị phân (Binary operators) hoạt động trên hai toán hạng** (được gọi là trái và phải). Một ví dụ về toán tử nhị phân là toán tử +. Ví dụ, được cho 3 + 4, toán tử + lấy toán hạng bên trái (3) và toán hạng bên phải (4) và áp dụng phép toán bổ sung để tạo ra giá trị đầu ra mới là 7. Toán tử chèn (<<) và trích xuất (>>) là toán tử nhị phân, lấy std :: cout hoặc std :: cin ở bên trái để xuất hoặc biến thành đầu vào ở bên phải.

**Toán tử ternary (Ternary operators)** hành động trên ba toán hạng. Cái này chúng ta sẽ trình bày sau.

Lưu ý rằng một số toán tử có nhiều hơn một nghĩa tùy thuộc vào cách chúng được sử dụng. Ví dụ, toán tử – có hai bối cảnh. Nó có thể được sử dụng ở dạng đơn nhất để đảo ngược một số (ví dụ: để chuyển đổi 5 thành -5 hoặc ngược lại) hoặc có thể được sử dụng ở dạng nhị phân để thực hiện phép trừ (ví dụ: 4 – 3).

**3. Chuỗi operators**

Các toán tử(**operators**) có thể được nối với nhau sao cho đầu ra của một toán tử có thể được sử dụng làm đầu vào cho một toán tử khác. Ví dụ, được đưa ra như sau: 2 \* 3 + 4, toán tử nhân đi trước và chuyển đổi toán hạng trái là 2 và toán hạng phải là 3 thành giá trị mới là 6 (trở thành toán hạng bên trái cho toán tử cộng). Tiếp theo, toán tử cộng thực thi và chuyển đổi toán hạng trái là 6 và toán hạng phải là 4 thành giá trị mới là 10.

Chúng ta sẽ nói nhiều hơn về thứ tự cuả các toán tử về sau này. Cho đến nay, nó đủ để biết rằng các toán tử số học thực hiện theo thứ tự giống như chúng làm trong toán học chuẩn: Dấu ngoặc đơn trước, sau đó là Số mũ, sau đó là Phép nhân & Phép chia, rồi Phép cộng & Phép trừ.