CHAPTER 01. C++ 첫걸음

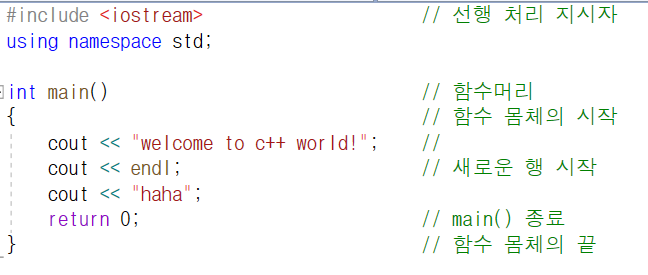
## C언어

* 절차지향적 언어이다.
* 하드웨어를 효율적으로 제어할 수 있는 간결한 언어
* 절차지향적 언어는 프로그래밍에서 데이터보다 알고리즘에 더 치중한다는 뜻
* 절차지향적 언어는 모듈 구성이 용이하며 구조적인 프로그래밍이 가능하다.
* 컴퓨터의 처리구조와 유사해 실행속도가 빠르다.
* 유지보수가 어렵고 대형 프로젝트에 부적합하다.
* 코드가 길어지면 가독성이 무척 떨어지며 이해하기 힘들다.

## C++

* 객체지향적 언어이다.
  + 객체: 추상화의 산물.
  + 추상화: 변수나 메소드들을 캡슐화하고 그것을 외부에서 사용 못하게 information hiding하는 과정을 말한다.
* 객체 지향 프로그래밍은 데이터를 강조하므로 언어 자체를 해결해야 할 문제에 맞춘다. 즉, 해결해야 할 문제의 특성에 맞게 데이터형 자체를 설계한다.

CHAPTER 02. C++ 시작하기



## main() 함수

### 인터페이스 역할을 하는 함수 머리

: 함수 머리는 호출 함수와 피호출 함수의 인터페이스를 나타낸다. 함수 이름의 앞 부분을 함수 리턴형(return type)이라고 한다. 이것은 피호출함수가 호출 함수로 다시 넘겨주는 정보의 흐름을 나타낸다. 함수 이름 뒤에 있는 괄호 안의 부분을 인자 리스트(argument list) 또는 매개변수 리스트(parameter list)라고 한다.

이것은 호출 함수가 피호출함수로 넘겨주는 정보의 흐름을 나타낸다.

: 다른 함수에 의해 호출된 함수는 자신을 호출한 함수에게 값을 리턴할 수 있다. 이 값을 리턴값(return type)이라고 한다.

: main 함수의 함수머리 ‘int main()’은 main()과 OS 사이의 인터페이스를 나타낸다.

: int main(void) 라고 해도 되고 void main(void)라고 해도 된다. 그러나 void 리턴형은 일부 시스템에서는 동작하지 않는다. C++ 표준이 아니기 때문이다.

: 모든 C++ 프로그램에는 메인함수가 반드시 하나 있어야 한다. 그러나 DLL모듈을 작성할 때와 같이 독립된 프로그램이 아니면 메인 함수가 필요 없다. 로봇의 컨트롤러 칩과 같은 특수한 황경에서도 필요하지 않을 수도 있다. 그러나 일반적인 독립형 프로그램에는 반드시 메인함수가 필요하다.

## C++ 전처리기와 iostrem 파일

: C와 마찬가지로 C++도 전처리기(preprocessor)를 사용한다.

: 전처리기는 컴파일을 하기 전에 소스 파일에 대해 미리 어떤 처리를 수행하는 프로그램이다.

: #include 전처리 지시자를 보통 사용한다. 이 지시자는 전처리기에게 특정 파일의 내용을 프로그램에 추가하라고 지시한다. 이와 같이 컴파일되기 전에 소스 코드에 텍스트를 추가하거나 텍스트를 대체하는 것이 전처리기가 수행하는 기본적인 역할이다.

: C++ 헤더파일에는 확장자를 사용하지 않기로 했다. 그리고 c 헤더파일의 이름에서 .h를 뺀 것이다. 예를 들어 math.h 헤더파일을 C++에서는 츰소로 이름이 바뀌었다.

## namespace

using 지시자를 사용해서 이름 공간(namespace)에 들어있는 모든 이름을 사용할 수 있다.

*using namespace std;*

를 사용하여 std::접두어를 사용하지 않고도 std 내 메소드를 사용할 수 있다.

## cout을 사용한 C++의 출력

cout은 문자열(string), 수(number), 문자(character)들을 포함한 여러 가지 다양한 정보들을 출력하는 방법을 알고 있는 미리 정의된 객체이다.

출력은 프로그램으로부터 흘러나오는 일련의 문자들, 즉 스트림이다. cout 객체가 이 스트림을 나타낸다. cout 객체의 속성은 iostream 파일에 정의되어 있다. cout 객체의 속성은 삽입 연산자(<<)를 포함하고 있다

* 추출 연산자 (>>): 데이터를 입력스트림에서 추출
* 삽입 연산자 (<<): 데이터를 출력스트림에 삽입

## 조정자 endl

두이은 출력 스트림에 삽입하면 화면 커서가 다음 행의 시작 위치로 간다. cout에게 특수한 의미를 가지는 두이과 같은 특별한 표기들을 조정자(munipulator)라고 한다.

## 개행문자 \n

\n 조합은 개행 문자(newline character)라고 부르는 하나의 문자를 나타내는 것으로 간주된다.

## 함수

C++ 프로그램은 함수라고 부르는 하나 또는 그 이상의 모듈들로 이루어진다. 프로그램은 main()이라는 함수부터 실행이 된다. 그러므로 어떤 프로그램이든지 간에 main() 함수가 반드시 하나 있어야 한다. 함수는 함수 머리와 함수 몸체로 이루어진다. 함수 머리는 그 함수가 어떤 데이터형의 값을 리턴하고 어떤 종류의 정보를 매개변수로 받는지를 나타낸다. 함수 몸체는 중괄호 { } 로 둘러싸인 C++의 구문으로 이루어진다.

### C++ 구문 유형

선언구문(declaration statement)

* 함수에서 사용되는 변수의 이름과 데이터형을 선언한다.

대입구문(assignment statement)

* 대입 연산자를 사용하여 변수에 값을 대입한다.

메시지구문(message statement)

* 객체에 메시지를 전달하여 활동을 시작하게 한다.

함수호출(function call)

* 함수를 동작시킨다. 피호출 함수가 종료되면 프로그램은 호출 함수의 함수 호출 바로 뒤에 있는 구문으로 복귀한다.

함수원형(function prototype)

* 함수가 기대하는 매개변수의 개수, 매개변수의 데이터형, 함수의 리턴형을 선언한다.

리턴형(return statement)

* 피호출 함수가 리턴하는 값을 호출 함수에 전달한다.

