Introduction to C++ <chapter 2>

* C++ Fundamental type: 6 categories
* int: integer 32bit
* double: Floating point 64 bit
* char: Character
* bool: Boolean
* void: no value type
* nullptr: Null pointer
* C++ provides several integer types, with different ranges of values and memory sizes
* short, long, signed short/long, unsigned short/long
* Interger Literals
* octal(8진법) 0을 붙임

017 = 1\*8+7 = 15, 0177 = 1\*64 + 7\*8 + 7 = 127

* Hexadeciamal(16진법) 0x를 붙임

0~9 mean 0~9, a~f (or A~F) mean 10, 11, 12, 13, 14, 15

0xf = 1\*15 = 15

0xff = 15\*16 + 15\*1 = 255

* Floating point types
* float, double, long double
* 0. is ‘double 0’ / 0 is ‘int 0’
* 123e6 = 123.0 \* / 0.123e12 = 0.123 \*
* .0 is ‘double 0.0’
* Character types
* A char is intended to store any character in the computer’s built-in character set.
* one byte of memory: Signed and unsigned char are typically used to store really short integers: **signed char** typically ranges from -128 to 127 and **unsigned char** typically ranges from 0 to 255 (1byte = 8 bits (bit 0 or 1), so 2^8 = 256)
* So, char is either signed or unsigned, depending on the system

Always safe to store values from 0 to 127 in a char

* ‘wchar\_t’: wide character type / ‘char16\_t’ is 16 bits wide, for UTF-16 / ‘char32\_t’ is same
* 아스키 코드로 해석이 된다
* The Boolean type
* the bool type is part of the integer types: Can be used in arithmetic calculations.
* The void type
* The void type says “no value” and “no memory is used”
* void is used as the return type of a function does not return any value -> simple functions.
* Null type
* Integer division and modulus: left to right
* 6/4 = 1(몫), 6%4 = 2(나머지), -7/2 = -3 ( ‘-3’ \* 2 + ‘-1’ ) 이렇게 스플릿 한다
* Integer promotions
* bool, char, short는 아스키코드로 변환이 가능하고, 산술적 계산이 가능하다

ex) true + 'a' - 5 evaluates to 93

* Usual Arithmetic conversions
* Integer types and floating types can be intermixed in arithmetic expressions.
* The result of the expression is of the “bigger” type. (대부분 정수 -> 실수이고, 실수 내에서도 double, long double 형태가 가장 강한 타입이다.)
* 실수 나눗셈음 소수점도 표현하지만, 정수끼리의 나눗셈은 몫만 표현한다.
* Local automatic variables
* A function (or any code block) may declare and use local automatic variable
* cin
* Standard input stream object (python의 input)