**- OPERATOR –**

|  |
| --- |
| **Operator:** Special symbols that perform specific operations on operands, and then return a result. |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Precedence | Type | | Operator | Description | Associativity |
| 1 | First Operator | | ( ) | Function call | → |
| 2 | Unary Operator | | ! | Logical negation (opposite) | ← |
| ++ x | Prefix increment: increments the value of x, and then returns the incremented value.  x = 1  y = ++x;  *output: x is 2, y is 2* |
| x++ | Postfix increment:  increments the value of x, but returns the original value that x held before being incremented.  x = 1  y = x++;  *output: x is 2, y is 1* |
| -- | Postfix/prefix decrement  (Works the same as the increments) |
| 3 | 이  항  연  산  자 | Multiplicative Operator  (산술연산) | \* | Multiplication | → |
| / | Division |
| % | Modulo: Returns the remainder of the two numbers after division |
| 4 | Additive Operator  (산술연산) | + | Addition |
| - | Subtraction |
| 5 | Relational Operator  (비교연산) | < | Less than |
| <= | Less than (inclusive) |
| > | Greater than |
| >= | Greater than (inclusive) |
| == | Equality |
| != | Inequality |
| 6 | Bitwise AND  (비트연산) | & | Binary AND Operator copies a bit to the result if it exists in both operands.  a = 0011 1100  b = 0000 1101  *(A & B) will give 12 which is 0000 1100* |
| 7 | Bitwise OR  (비트연산) | | | Binary OR Operator copies a bit if it exists in either operand.  (A | B) will give 61 which is 0011 1101 |
| 8 | Logical AND  (논리연산) | && | expression1 && expression2  *true only if both expression1 and expression2 are true* |
| Logical XOR | ^ | If only **one** of the expressions are true  int a =10;  int b = 5;  a >b ^ a = 10 → false  a >b ^ a = b → true |
| Logical NOT | ! | The opposite value  int a =10;  int b = 5;  (! (a > b)); → false  (! (a < b)); → true |
| 9 | Logical OR  (논리연산) | || | expression1 || expression2  *true if either expression1 or expression2 is true* |
| 10 | Conditional Operator  (Ternary Operator) | | ? : | variable = **Expression ? expression1 : expression2**  If the Expression is true, expression1 is assigned to the variable.  If the Expression is false, expression2 is assigned to the variable. |
| 11 | Assignment Operator  (대입연산) | | = | a = b → a = b; | ← |
| += | a += b → a = a+ b; |
| -= | a -= b → a = a - b; |
| \*= | a \*= b → a = a \* b; |
| /= | a /= b → a = a / b; |
| %= | a %= b → a = a % b; |