

C Programming (W6)



Welcome!!

Please check attendance individually.
(Mobile App)

Things to do today

- 01** | Lecture Notes (Ch.4 ~ Ch.5)
- 02** | Check the homework link
 - Everyone put your ID & name at the top of readme.md
- 03** | About midterm
 - The midterm exam will be replaced by a mini-project.
 - The mini-project will be given approximately 2-3 weeks later.

Things to do today

Expression:

An **expression** is a combination of variables, constants, operators, and function calls that are evaluated to produce a value.

Example: `a + b` is an expression.

Operation: (Ch.4)

An **operation** is the action performed by an operator on one or more operands.

Example: In `a + b`, the `+` operator performs an addition operation.

Statement: (Ch.5 ~ Ch.6)

complete instruction that tells the computer to do something.

Example: `int sum = a + b;`

Statement

In **C**, a **statement** is a complete instruction that tells the computer to do something. It usually ends with a **semicolon (;)**.

Think of a statement as a **single action** or command in a C program.

A statement is the basic unit of a program.

Compound Statement / Block. { }

Types of Statements in C:

1. **Declaration Statement:** Used to declare variables.

```
int a;
```

```
float pi = 3.14;
```

1. **Control Flow Statements:** Control the flow of execution.

- **Conditional statement:**

```
if (a > b) { ... }
```

- **Iteration Statement (Looping):**

```
for (int i = 0; i < 10; i++) { ... }
```

3. **Jump Statements:** Change the flow directly.

```
break;
```

```
continue;
```

```
return;
```

```
goto
```

See you next week!

DO NOT miss the classes



Expression (식):

식은 변수, 상수, 연산자, 함수 호출 등을 조합하여 하나의 값을 생성하는 코드 조각입니다.

예시: `a + b`는 하나의 식입니다.

Operation (연산): (Ch.4)

연산은 연산자가 하나 이상의 피연산자에 대해 수행하는 동작입니다.

예시: `a + b`에서 `+` 연산자는 덧셈 연산을 수행합니다.

Statement (문): (Ch.5 ~ Ch.6)

문은 컴퓨터에게 어떤 작업을 수행하라고 지시하는 완전한 명령입니다.

예시: `int sum = a + b;`는 하나의 문입니다.

1. Arithmetic Operators (산술 연산자)

기본적인 수학 연산을 수행할 때 사용됩니다.

예: $+$, $-$, $*$, $/$, $\%$ 등

2. Relational (Comparison) Operators (관계(비교) 연산자)

두 값을 비교할 때 사용됩니다.

예: $==$, $!=$, $>$, $<$, $>=$, $<=$ 등

3. Logical Operators (논리 연산자)

불리언(boolean) 표현식을 결합하거나 반전시킬 때 사용됩니다.

예: $\&\&$, $\|\|$, $!$ 등

4. Assignment Operators (대입 연산자)

변수에 값을 할당할 때 사용됩니다.

예: $=$, $+=$, $-=$, $*=$, $/=$, $\%=$ 등

Statement

In **C**, a **statement** is a complete instruction that tells the computer to do something.

It usually ends with a **semicolon (;)**. Compound Statement / Block. { }

Types of Statements in C:

1. **Declaration Statement:** Used to declare variables.

```
int a;
```

```
float pi = 3.14;
```

2. **Control Flow Statements:** Control the flow of execution.

- **Conditional statement:**

```
if (a > b) { ... }
```

```
else {...}
```

- **Iteration Statement (Looping):**

```
for (int i = 0; i < 10; i++) { ... }
```

```
while() {...}
```

- Cooking... (Entry → End)

Statement

C에서 문장은 컴퓨터에게 어떤 작업을 하라고 지시하는 완전한 명령어.

일반적으로 세미콜론(;)으로 끝남.

복합 문장/블록은 { } 사용.

C의 명령문 유형:

1. 선언문: 변수를 선언하는 데 사용.

```
int a;
```

```
float pi = 3.14;
```

2. 제어 흐름 문: 실행 흐름을 제어.

- 조건문

```
if (a > b) { ... }  
else {...}
```

- 반복문

```
for (int I = 0; I < 10; i++) { ... }  
while() {...}
```