

# 1강. 프로그래밍 언어? C언어?가 뭐야?

- 이용현 -

# 목차

1. 프로그래밍 언어?
2. 프로그래밍을 하기전에 잠깐!
3. C언어가 뭐야?

# 1. 프로그래밍 언어?

# 1. 프로그래밍 언어?

프로그래밍 언어 (Programming Language)

: 컴퓨터를 이용하여 특정문제를 해결하기 위한 프로그램을 작성하기 위한 언어

프로그래밍 언어의 종류

**C** **JAVA** **PYTHON**

저급언어(Low-level Languages) vs 고급언어(High-level Languages)

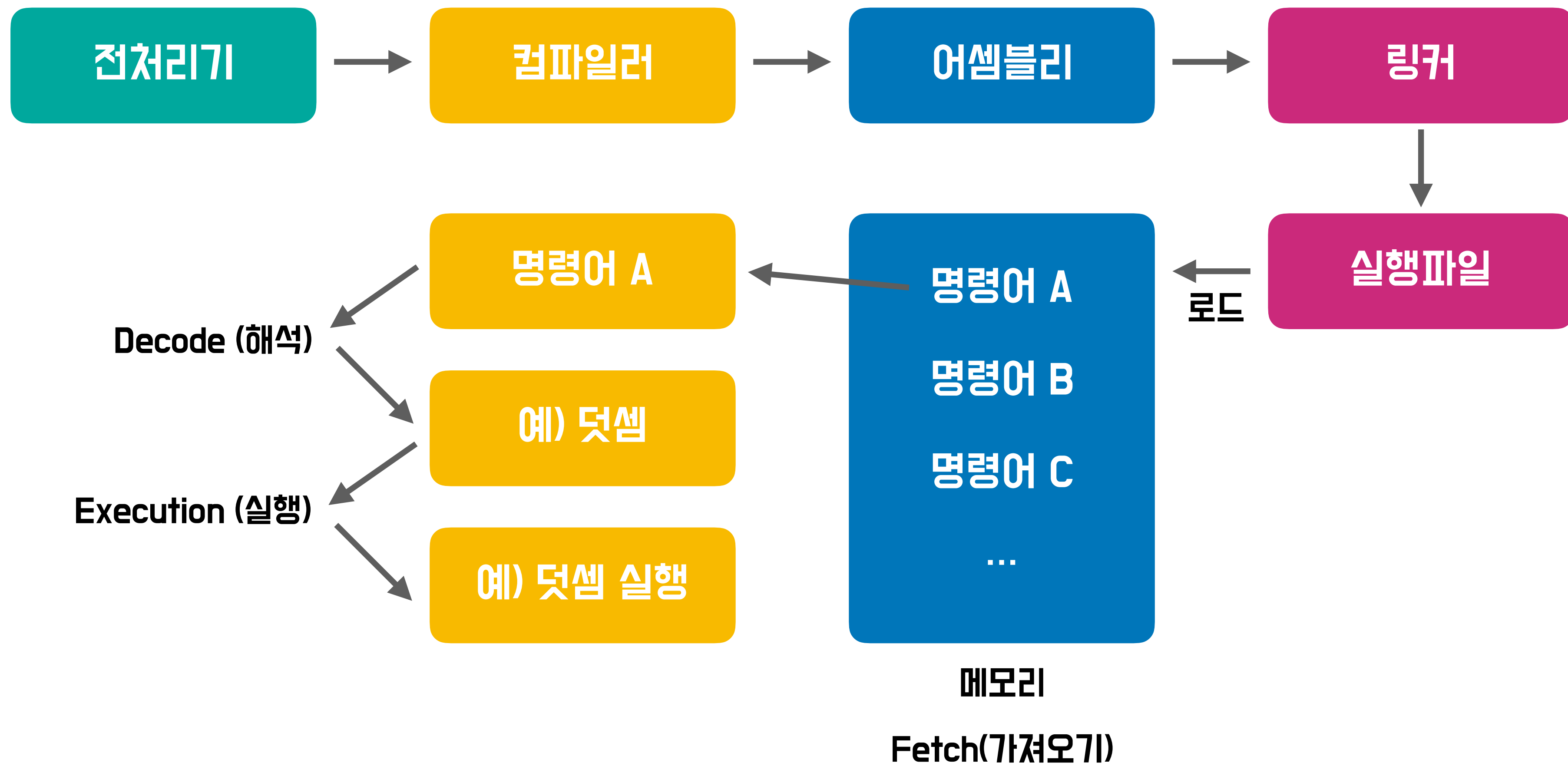
- 코드의 기능, 기술적인 부분이 모자라서 저급언어가 아니다.
- 저급언어에서 고급언어로 갈 수록 사람이 사용하는 언어의 문법과 유사율이 높다.
- 고급언어는 **\*컴파일러** 언어라고 하며, 하드웨어에 대한 지식이 없어도 작성하고 수정하기 용이하다.
- 예시 ) C, JAVA, PYTHON 등

## **\* 컴파일러 (Compiler)**

프로그램을 실행시키기 위해서는 '실행파일'이 필요하다. 실행파일을 만들 때 원시파일(Source file)을 컴퓨터가 알아 들을 수 있는 기계어 코드로 만들어야 한다. 이 동작을 컴파일(Compile)이라 부르며, 컴파일 작업을 하는 기능을 컴파일러라고 부른다.

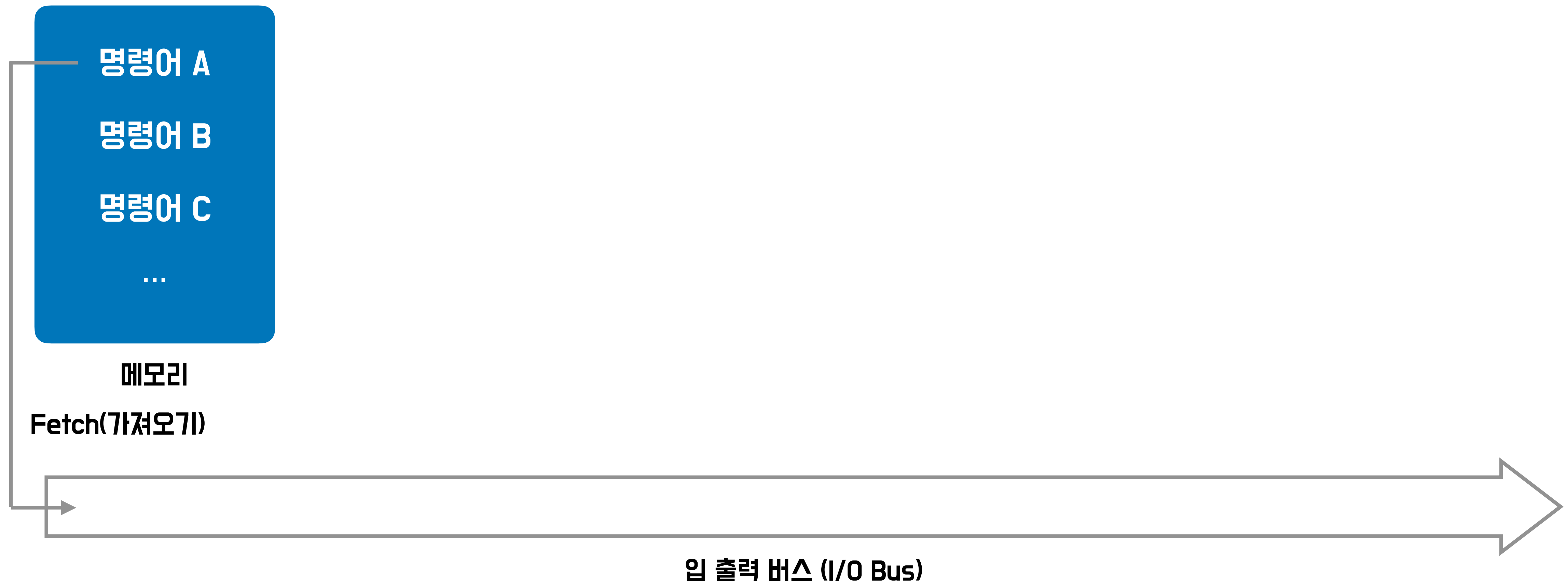
# 1. 프로그래밍 언어?

## 프로그램 구동원리



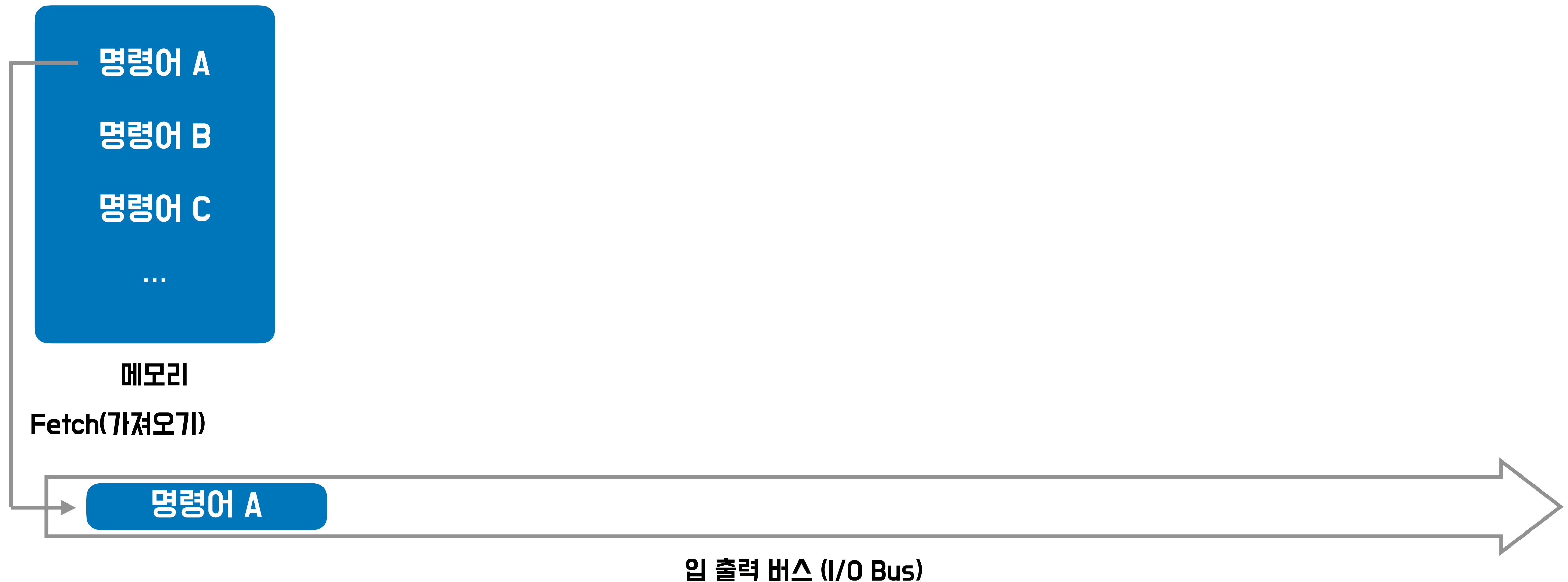
# 1. 프로그래밍 언어?

프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)



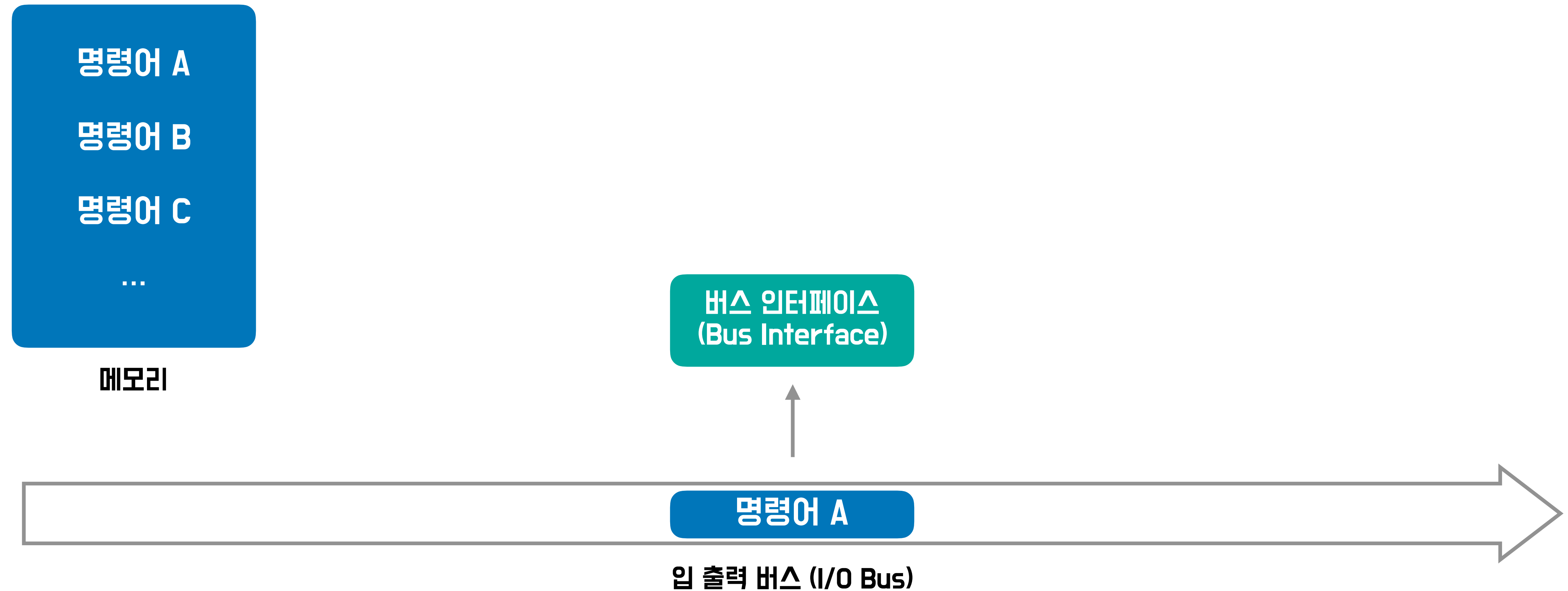
# 1. 프로그래밍 언어?

프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)



# 1. 프로그래밍 언어?

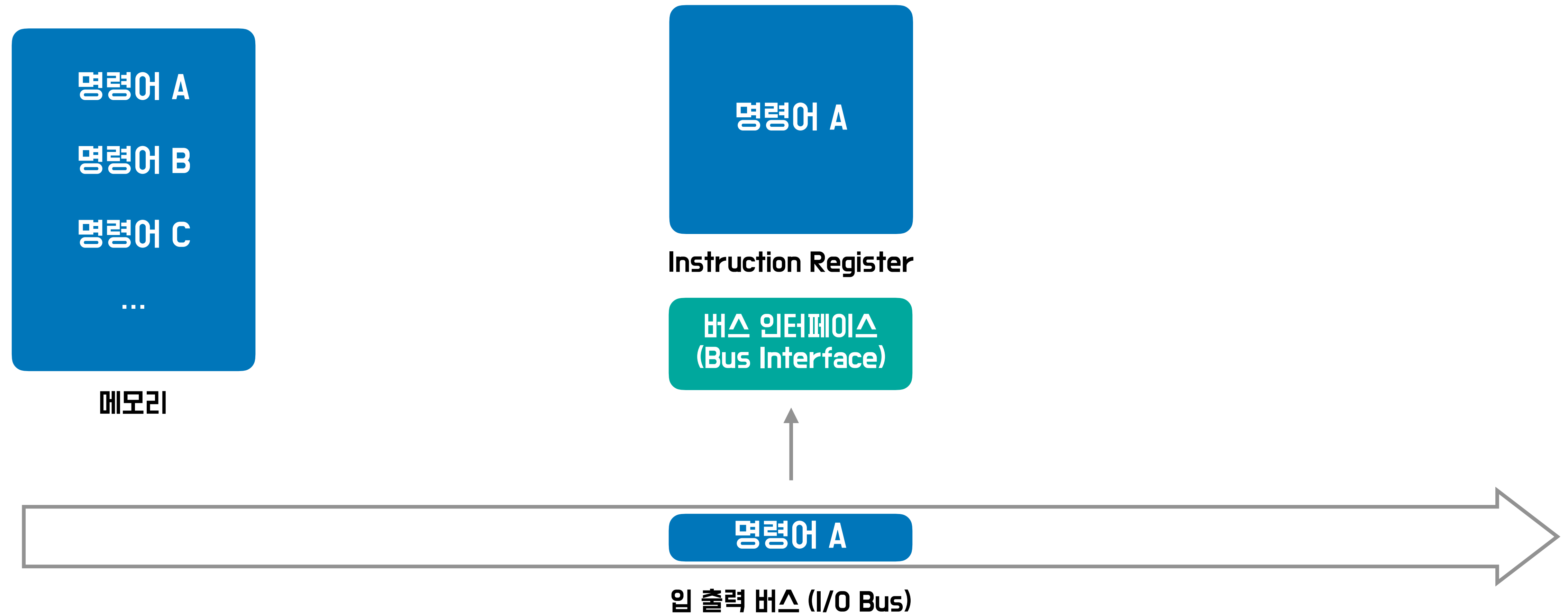
프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)





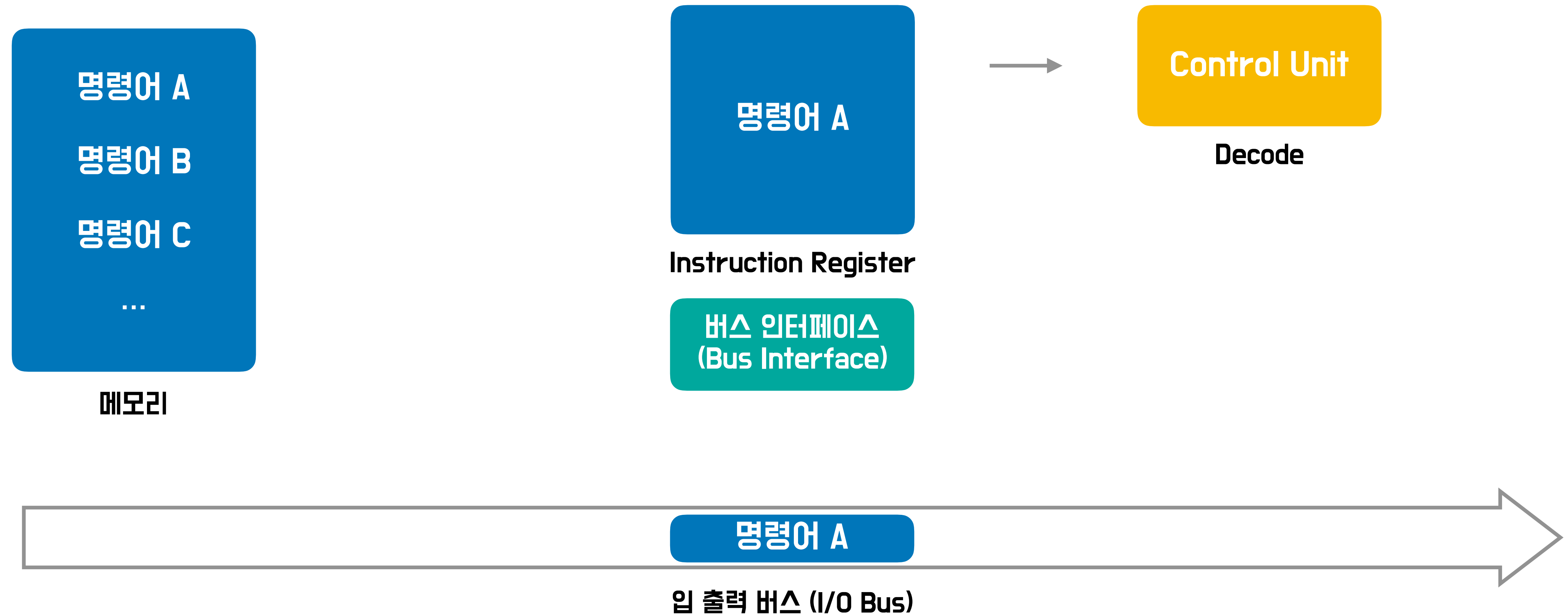
# 1. 프로그래밍 언어?

프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)



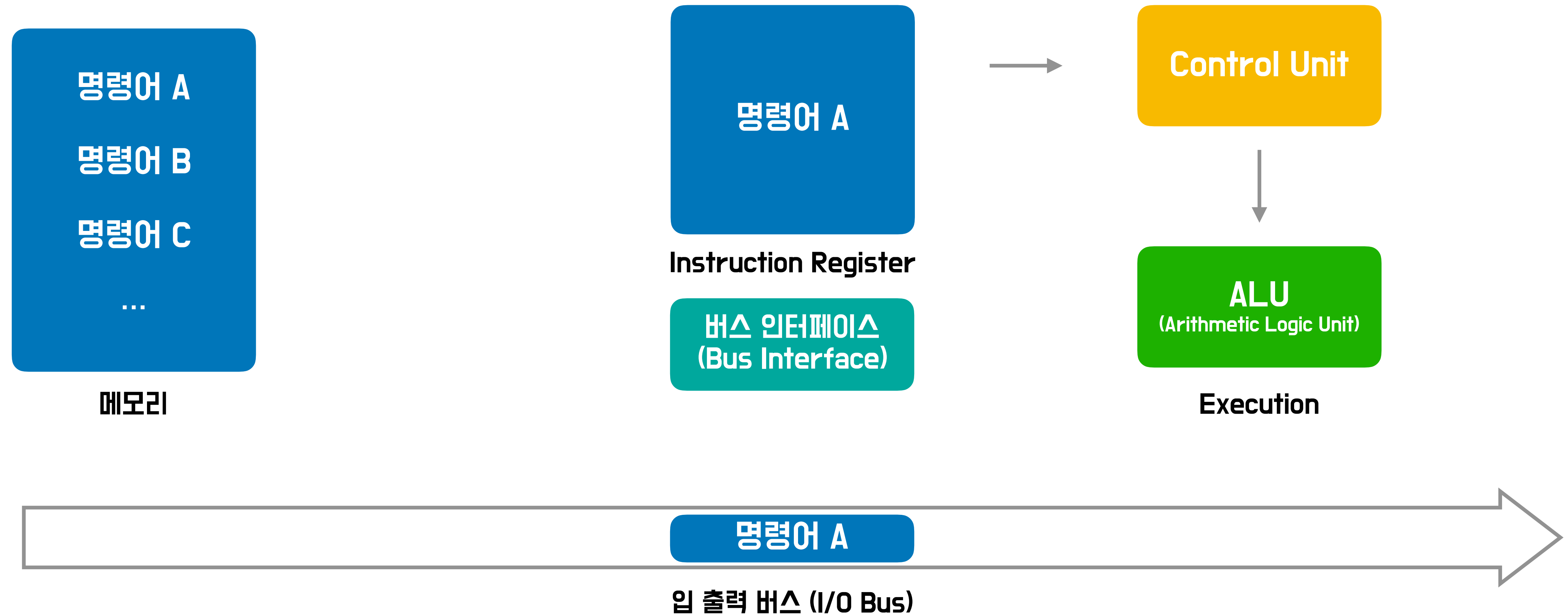
# 1. 프로그래밍 언어?

프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)



# 1. 프로그래밍 언어?

프로그램 구동원리 (CPU 관점에서 Fetch, Decode, Execution)



## 2. 프로그래밍을 하기전에 잠깐!

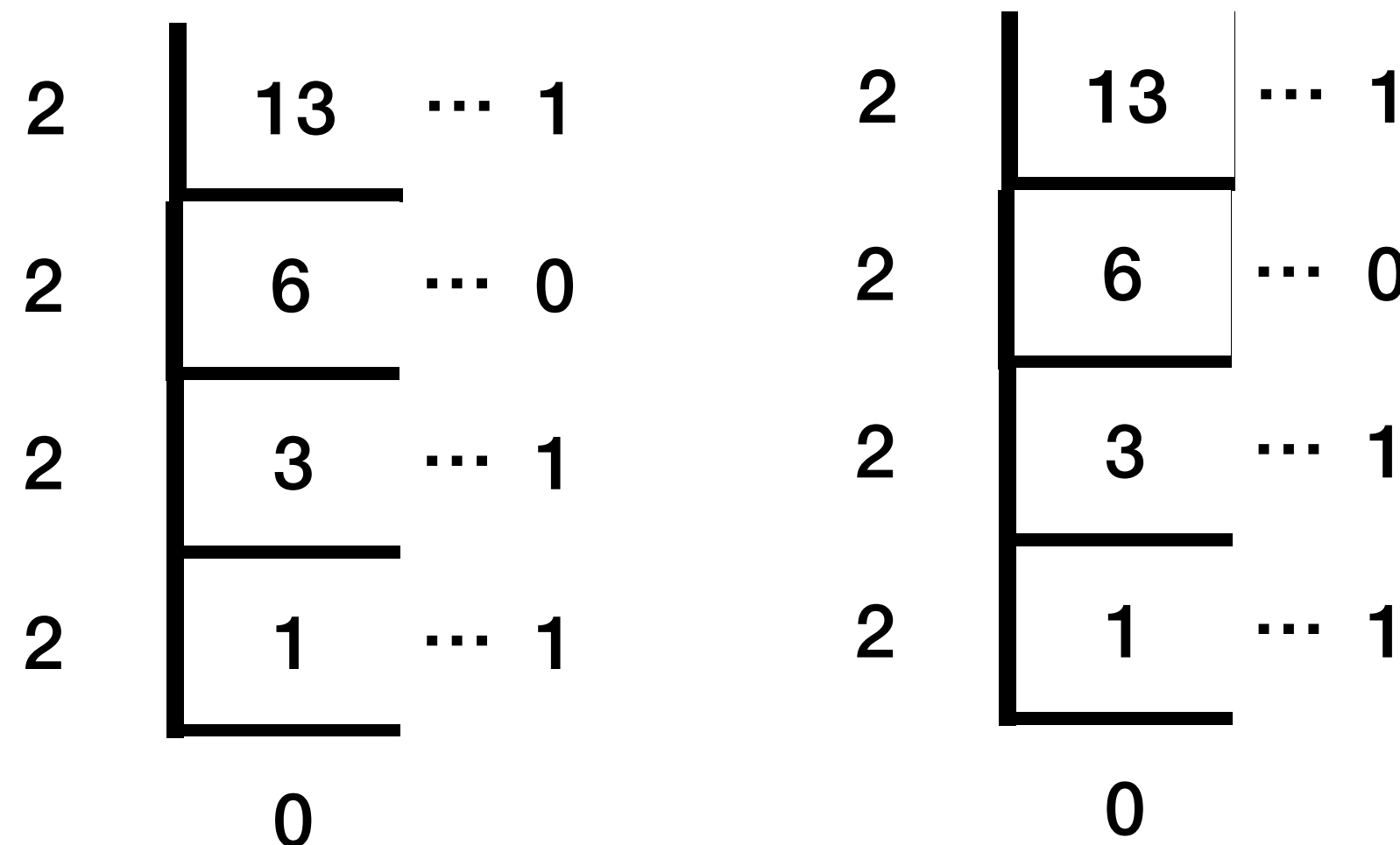
## 2. 프로그래밍을 하기전에 잠깐!

### 2진수?

- 우리가 일상생활 속에서 수를 셀 때 10진법을 이용한다. 10진법은 0~9의 수를 이용해서 수를 표현하는 방식이다.
- 2진수는 0~1까지의 수를 이용해서 수를 표현하는 방식이다.

### N진수 0 ~ (N-1)까지의 수를 이용해서 수를 표현하는 방식

- 10진수를 2진수로 표현하는 방법 (10진수 13을 2진수로 표현하기)



읽는 순서

표기방식

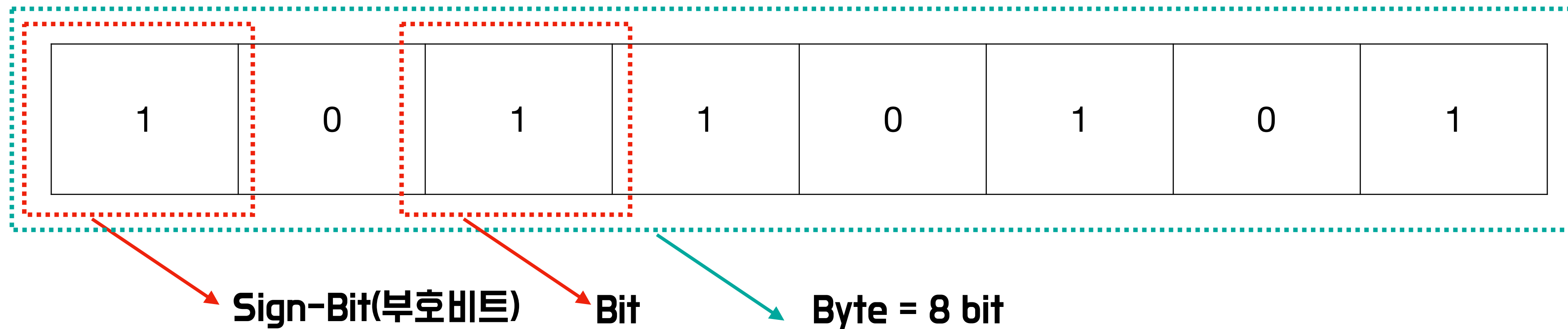
$$13_{(10)} = 1101_{(2)}$$

$$= 1 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 1 \times 2^0$$

## 2. 프로그래밍을 하기전에 잠깐!

### Bit와 Byte

- 컴퓨터에서 자료(Data)는 0(OFF, False)과 1(ON, True)로 이루어져있다.
- 0과 1 하나를 Bit라고 부르며, 8개의 Bit를 1 Byte라고 한다. - 맨 오른쪽 Bit는 부호비트이다. 0이면 +(양수), 1이면 -(음수)이다. (Unsigned data 제외)



8-bit로 나타낼 수 있는 숫자의 최대, 최소 범위

$$-2^8 \sim +(2^7 - 1)$$

$$= -2^8 \sim +(2^7 - 1)$$

n-bit로 나타낼 수 있는 숫자의 최대, 최소 범위

$$-2^n \sim +(2^{(n-1)} - 1)$$

### 3. C언어가 뭐야?

### 3. C언어가 뭐야?





### 3. C언어가 뭐야?

약 50년이 지난 구석기 시대 C언어는 왜 스테디셀러(Steady Seller)인가?

#### C언어가 왜 탄생했는가?

- C언어는 현재 운영체제(OS)들의 조상격인 UNIX OS를 쉽게 개발하기 위해 만들어졌다.
- C언어가 나오기 전에 UNIX OS는 '어셈블리 언어'라는 Low-level Language로 개발했다.
- CPU가 변화할 때마다 매번 서로 다른 어셈블리 언어로 작성해야하는 문제점이 있었다...
- 어떤 개발자들이 봐도 이해하고 수정할 수 있는 프로그래밍 언어의 필요 -> C언어 탄생

#### C언어를 왜 배워?

- 고급언어임에도 불구하고 하드웨어를 직접 제어할 수 있다.
- System programming을 하기 위해
- C언어를 기반으로 만들어진 C-Like Language가 많다. 예 ) JAVA, C++, C#, Objective-C 등

- 끝 -