

# 프로그래밍

춘천고등학교 소인수교육과정

C언어 Express

담당교수 : 안희국



# 이번 장에서 학습할 내용

- 프로그램의 이해
- 프로그래밍 언어
- 알고리즘
- 프로그램 개발 과정



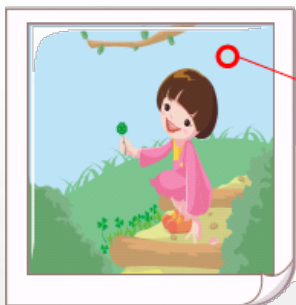
# 컴퓨터란?

**Q 컴퓨터(computer)는 무엇인가?**

**A** 컴퓨터는 기본적으로 계산(compute)하는 기계

**Q 컴퓨터를 이용하여 데이터를 처리하려면 반드시 데이터가 숫자 형태이어야 한다. 왜?**

**A** 컴퓨터는 숫자 계산을 하기 때문에 데이터는 숫자로 표시되어야 한다.



디지털 이미지의 경우

각 화소의 밝기와 색상을  
숫자로 표현  
한다,



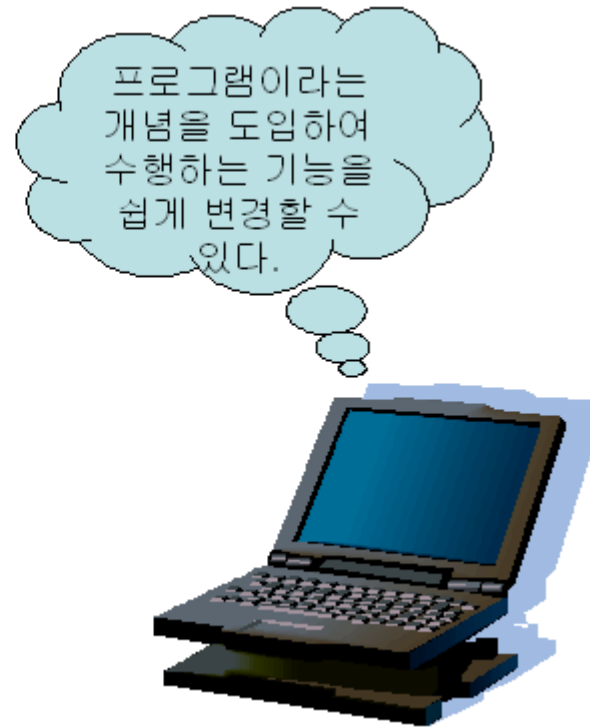
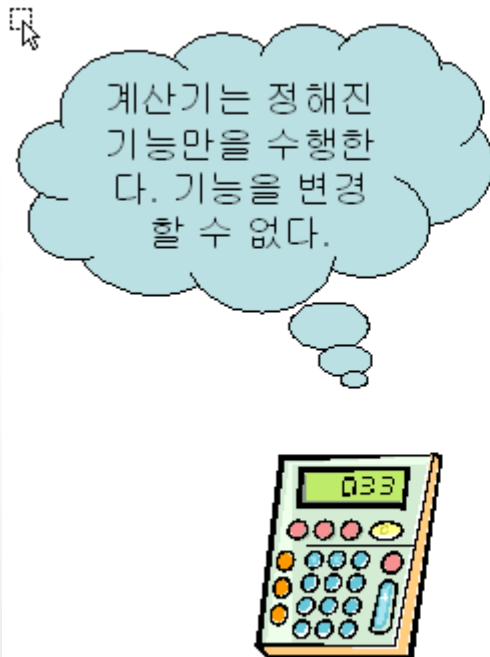
디지털 음악의 경우

각 파형의 높이를  
숫자로  
표현한다,

# 컴퓨터의 정의

**Q 그렇다면 계산만 빠르게 할 수 있으면 컴퓨터인가?**

**A 현대적인 의미에서의 컴퓨터는 명령어들의 리스트에 따라 데이터를 처리하는 기계라고 할 수 있다**



# 컴퓨터의 장점

**Q 컴퓨터의 가장 큰 장점은 무엇일까?**

**A** 컴퓨터는 범용적인 기계이다. 프로그램만 바꿔주면 다양한 작업이 가능하다.

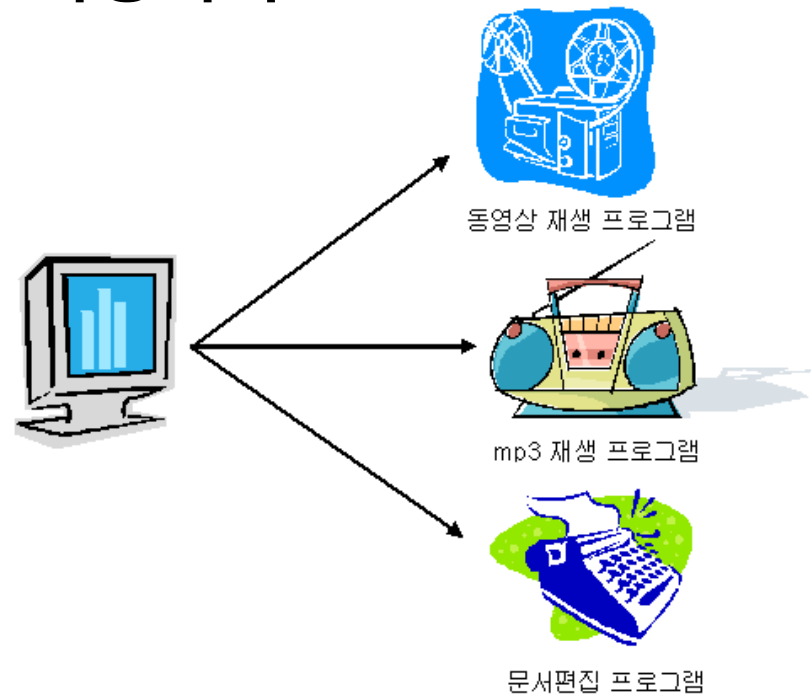


그림 1.1 프로그램만 바꾸어주면 컴퓨터는 다양한 작업을 할 수 있다.

# 스마트폰도 컴퓨터의 일종

A 피쳐폰과 다르게 애플리케이션만 변경하면 다양한 용도로 사용가능



# 컴퓨터의 구성 요소

**Q 컴퓨터의 구성 요소를 크게 2가지로 분류하면?**

**A** 컴퓨터는 기본적으로 하드웨어와 소프트웨어로 구분



하드웨어



소프트웨어

그림 1.2 하드웨어와 소프트웨어



# 하드웨어와 소프트웨어의 분리

- 최초의 컴퓨터에서는 분리되지 않았음



- 1950년대에 폰노이만이 제시한 아이디어
  - 프로그램을 메모리에 저장
  - 메모리에서 프로그램의 문장을 꺼내와서 하나씩 실행

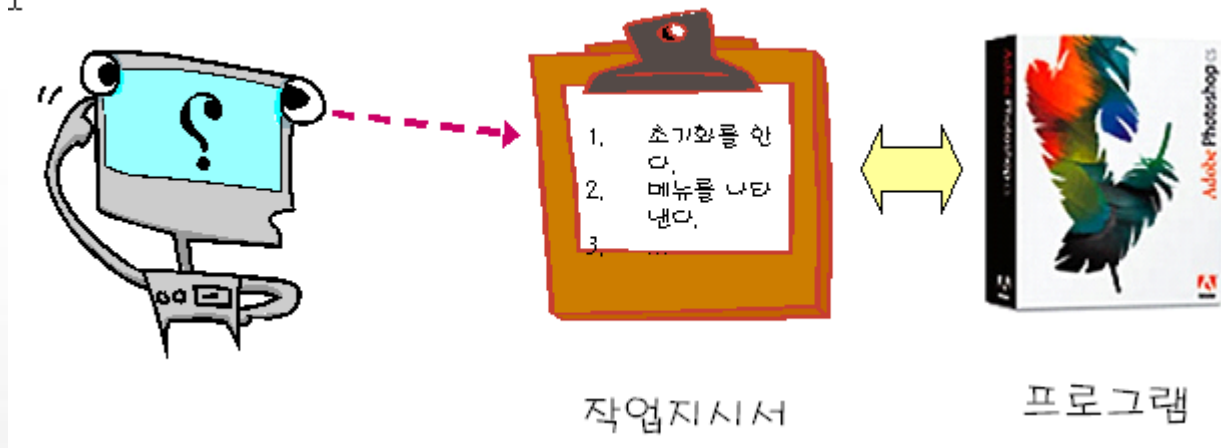


# 프로그램의 역할

**Q 컴퓨터에서 프로그램이 하는 일은 무엇인가?**

**A 프로그램이란 우리가 하고자 하는 작업을 컴퓨터에게 전달하여 주는 역할을 한다.**

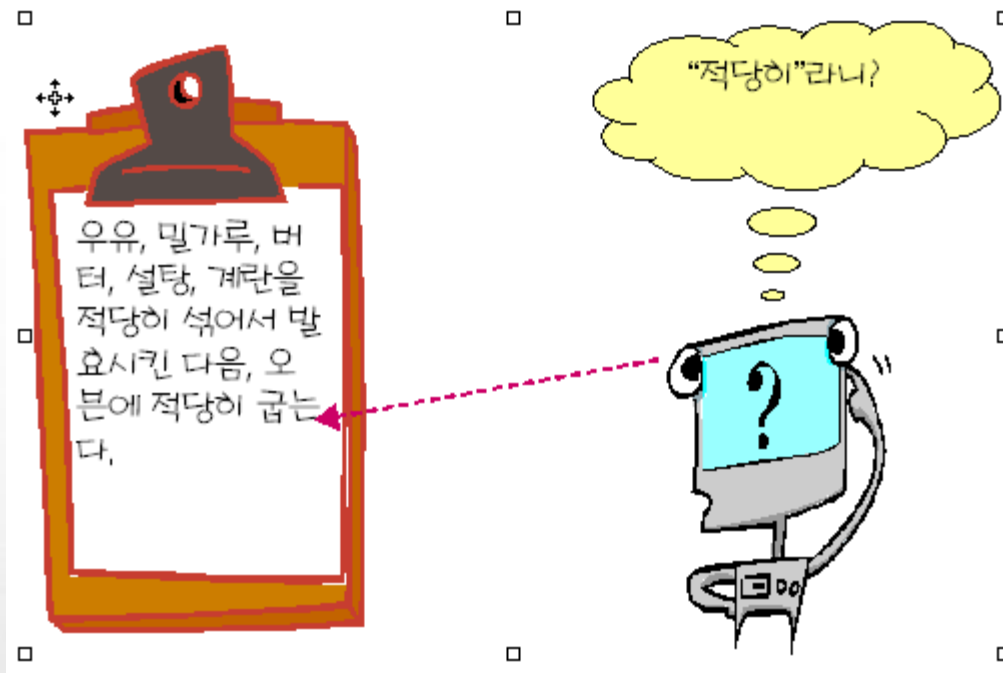
I



# 작업을 지시하는 방법

**Q 컴퓨터에게 적당히 작업을 시킬 수 있을까?**

**A 상식이나 지능이 없기 때문에 아주 자세하고 구체적으로 일을 지시하여야 한다.**





# 프로그램 안에 들어있는 것

- 컴퓨터에게 무엇을 어떻게 시킬 지를 문장으로 기록
- 프로그램의 각 문장은 컴퓨터에게 작업을 지시하는 명령(instruction)으로 되어 있다.

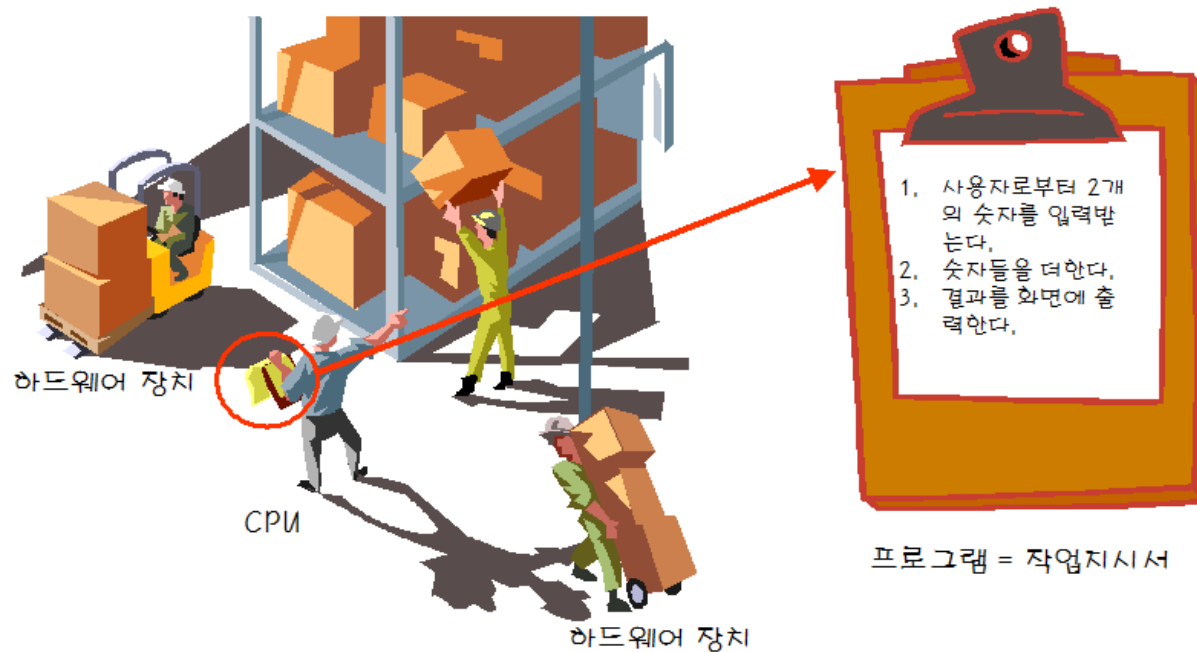


그림 1.4 프로그램은 작업 지시서와 같다.



# 이번 장에서 학습할 내용

- 프로그램의 이해
- 프로그래밍 언어
- 알고리즘
- 프로그램 개발 과정



# 컴퓨터가 이해하는 언어

- 컴퓨터는 인간의 언어를 이해할 수 없다.
- 컴퓨터는 이진수로 된 기계어만을 이해한다.

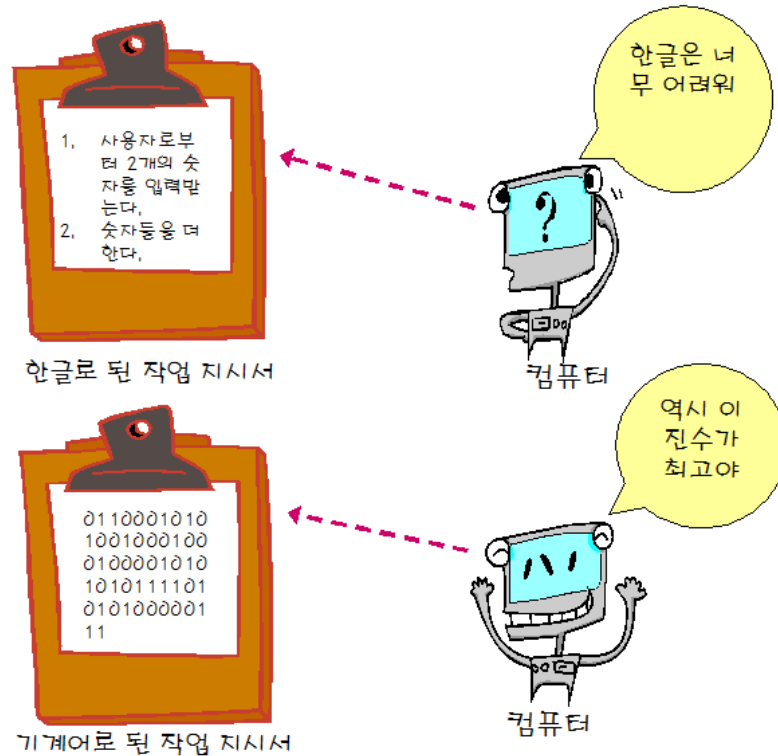


그림 1.5 컴퓨터는 한글로 된 작업 지시서는 이해하지 못하는 반면 기계어로 된 작업 지시서는 이해할 수 있다.

# 프로그래밍 언어의 역할

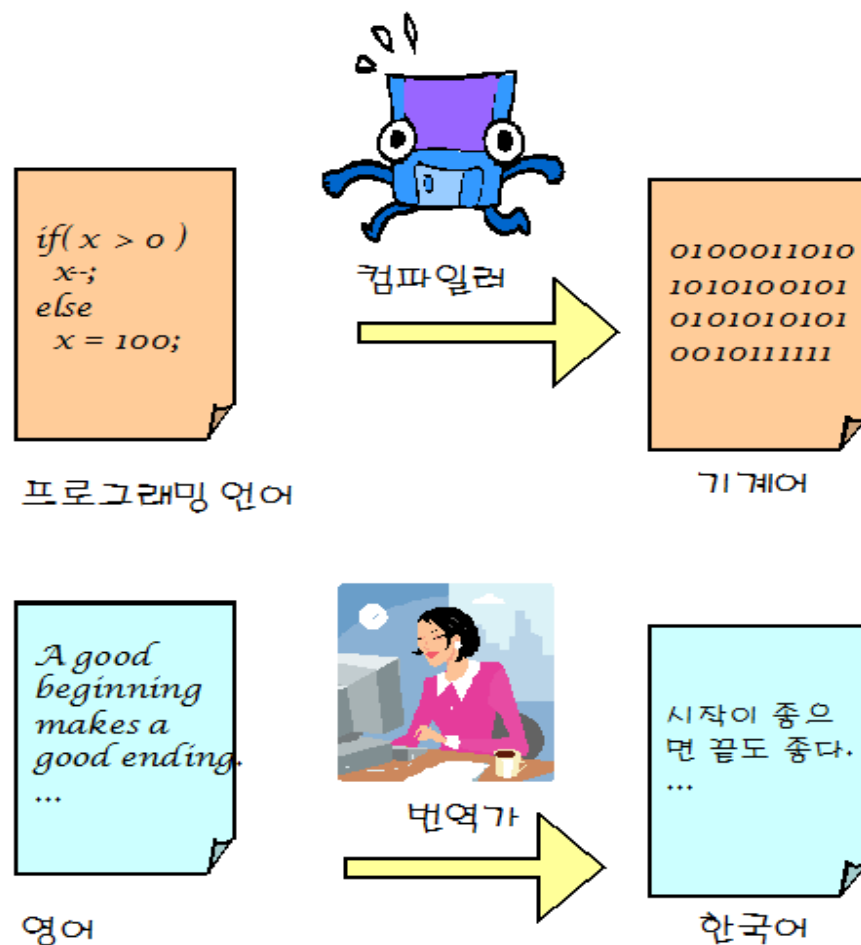
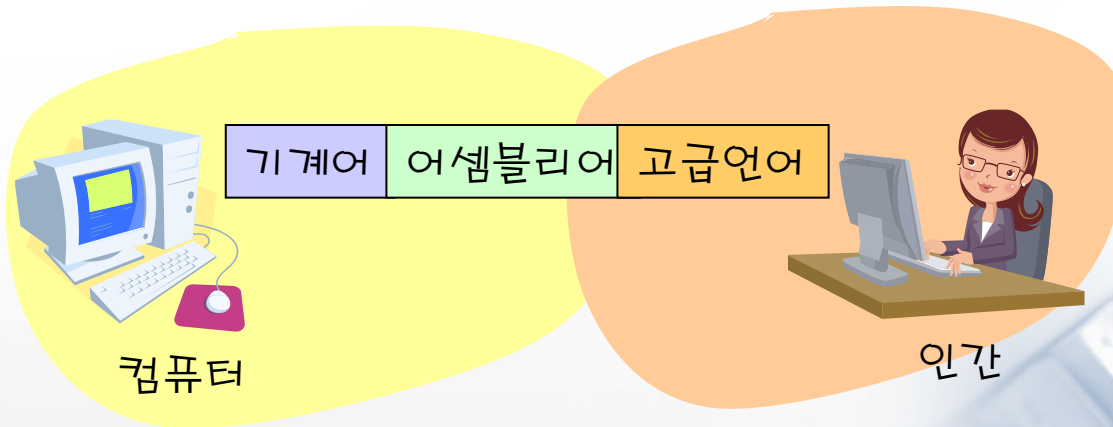


그림 1.6 컴파일러는 프로그램을 기계어로 변환한다.



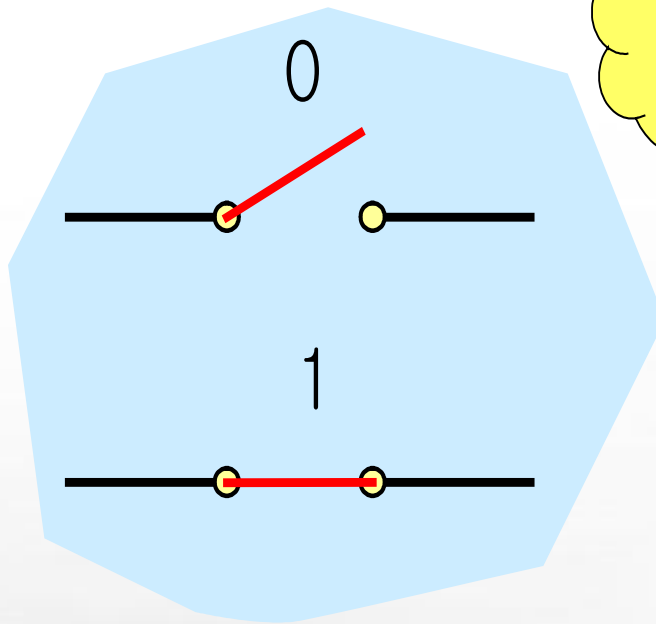
# 프로그래밍 언어의 분류

- 기계어(machine language)
- 어셈블리어(assembly language)
- 고급 언어(high-level language)



# 컴퓨터에 이진수가 사용되는 이유

- 이진수는 전자회로로 구현하기가 쉽다.



0은 열린 스위치로, 1은 닫힌 스위치로 표현할 수 있습니다.



- 1970년대 초 AT&T의 Dennis Ritchie 에 의하여 개발
- B언어->C언어
- UNIX 운영 체제 개발에 필요해서 만들어짐
- 처음부터 전문가용 언어로 출발



Ken Thomson과 Dennis Ritchie가 클린턴 대통령으로부터 National Medal of Technology 상을 받는 장면



# C언어의 버전

- K & R C

- 1978년 “C Programming Language“ 책 출간
- 비공식적인 명세서 역할

- ANSI C

- 1983년 ANSI(American National Standards Institute)는 X3J11이라는 위원회에 의한 표준

- C99

- 1999년에 ISO에 의한 표준
- C++에서 사용되는 특징 추가
- 아직 마이크로소프트는 지원하지 않음(이유: C++에 집중)

# C언어의 특징

- 간결하다.
- 효율적이다.
- C 언어는 하드웨어를 직접 제어하는 하는 저수준의 프로그래밍도 가능하고 고수준의 프로그래밍도 가능하다.
- C언어는 이식성이 뛰어나다.
- 초보자가 배우기가 어렵다.



# C언어의 특징

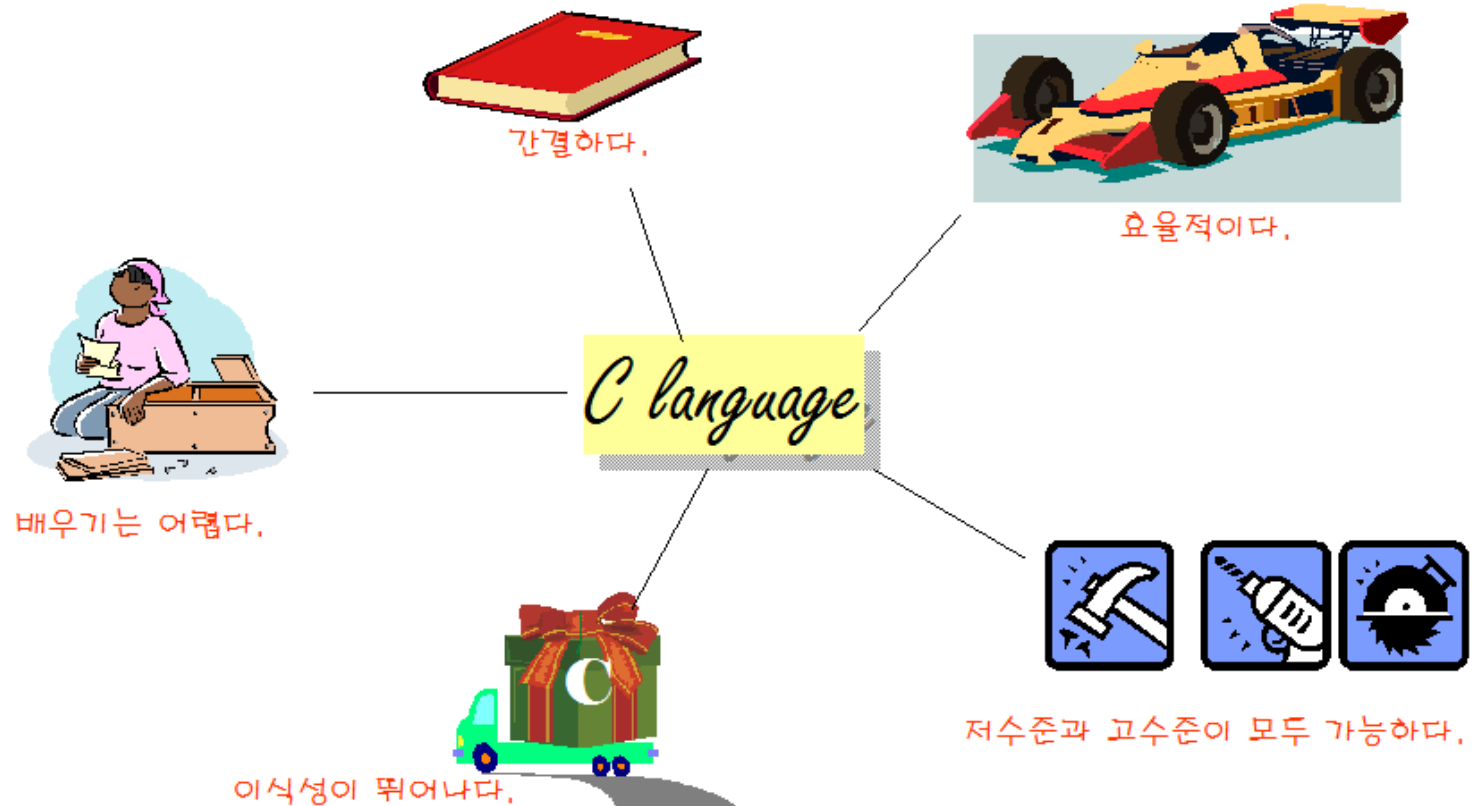


그림 1.10 C 언어의 특징



# C언어의 미래

**Q 앞으로도 C언어는 사용될 것인가?**

**A C언어는 C++와 JAVA, C#, Objective-C의 공통적인 부분이다.**

**A 임베디드 시스템에서는 C언어가 많이 사용된다.**

*임베디드 시스템: 임베디드 시스템이란 특수 목적의 시스템으로 컴퓨터가 장치 안에 MP3 플레이어, 스마트폰등이 여기에 속한다.*



mp3 플레이어도 CPU와 플래시 메모리 등이 들어가 있는 임베디드 시스템이다.



# 이번 장에서 학습할 내용

- 프로그램의 이해
- 프로그래밍 언어
- 알고리즘
- 프로그램 개발 과정



# 알고리즘

Q 오븐의 사용법만 배우고 음식 재료만 있으면 누구나 요리가 가능한가?

A 요리법을 알아야 한다.

프로그램이 요리와 같다면 알고리즘은 요리법에 해당한다.

알고리즘(algorithm): 문제를 해결하는 절차(방법)





# 빵을 만드는 알고리즘

- ① 빈 그릇을 준비한다.
- ② 이스트를 밀가루, 우유에 넣고 저어준다.
- ③ 버터, 설탕, 계란을 추가로 넣고 섞는다.
- ④ 따뜻한 곳에 놓아두어 발효시킨다
- ⑤ 170~180도의 오븐에서 굽는다



->



->



->



->



# 1부터 10까지의 합을 구하는 알고리즘

① 1부터 10까지의 숫자를 직접 하나씩 더한다.

$$I \quad 1 + 2 + 3 + \dots + 10 = 55$$

·② 두수의 합이 10이 되도록 숫자들을 그룹핑하여 그룹의 개수에 10을 곱하고 남은 숫자 5를 더한다.

$$(0 + 10) = 10$$

$$(1 + 9) = 10$$

$$(2 + 8) = 10$$

$$(3 + 7) = 10$$

$$(4 + 6) = 10$$

5

$$10 * 5 = 50 \quad + \quad 5 \quad = \quad 55$$

③ 공식을 이용하여 계산할 수도 있다.

$$10 * (1 + 10) / 2 = 55$$



# 알고리즘의 기술

- 순서도(flow chart): 프로그램에서의 논리 순서 또는 작업 순서를 그림으로 표현하는 방법

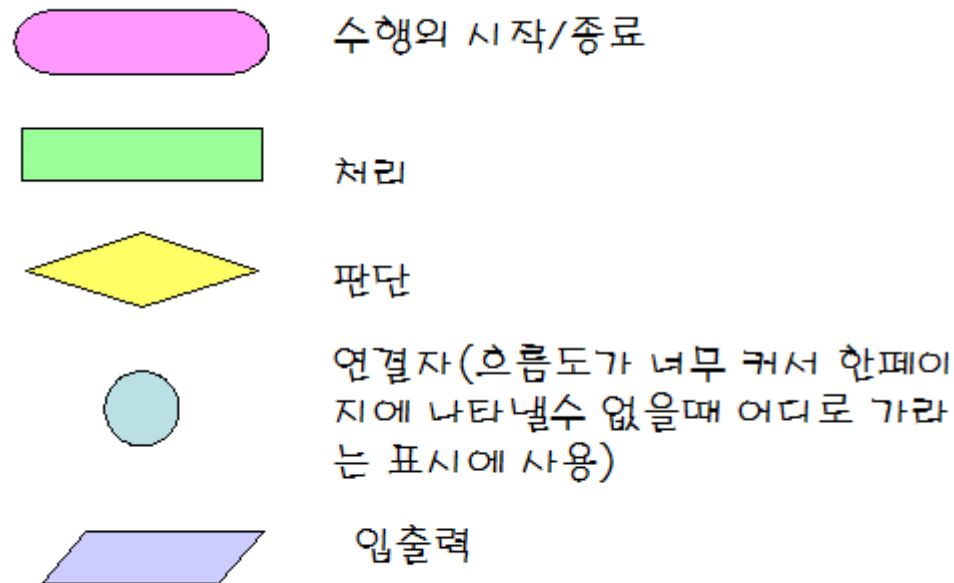


그림 1.13 순서도 작성에 사용되는 기호

# 알고리즘의 예

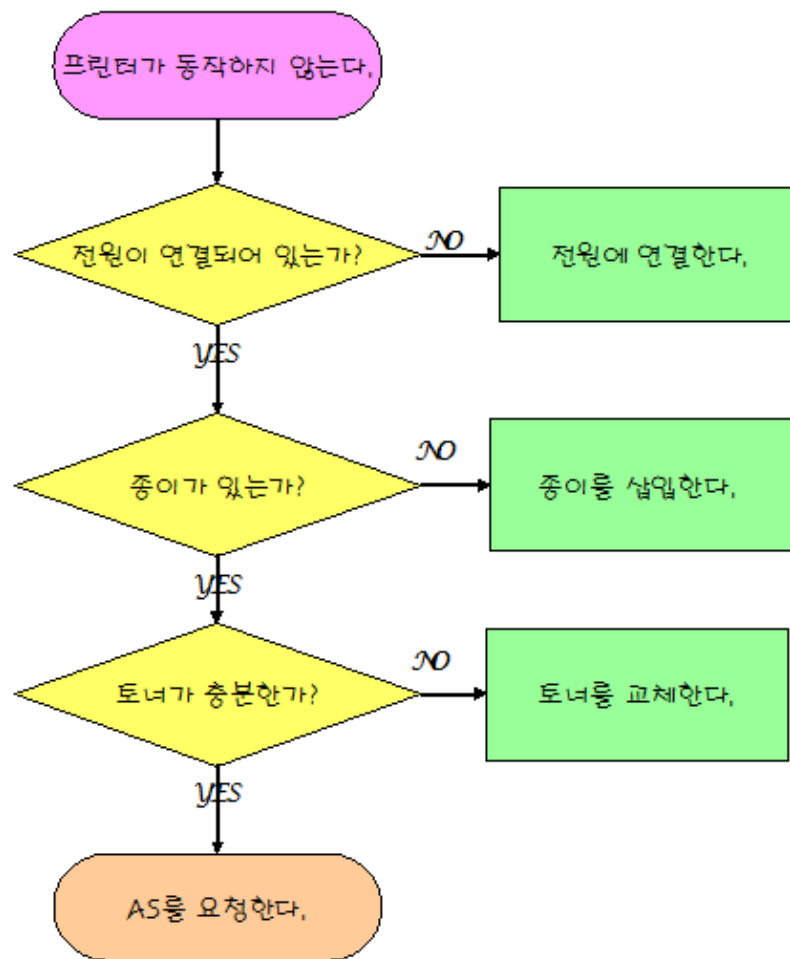


그림 1.14 순서도의 예: 프린터 고장을 처리하는 알고리즘

# 중간 점검

1. 친구에게 전화를 거는 알고리즘을 만들어보라.
2. 세탁기를 이용하여서 세탁을 하는 알고리즘을 만들어보라.





# 이번 장에서 학습할 내용

- 프로그램의 이해
- 프로그래밍 언어
- 알고리즘
- 프로그램 개발 과정



# 프로그램 작성 과정

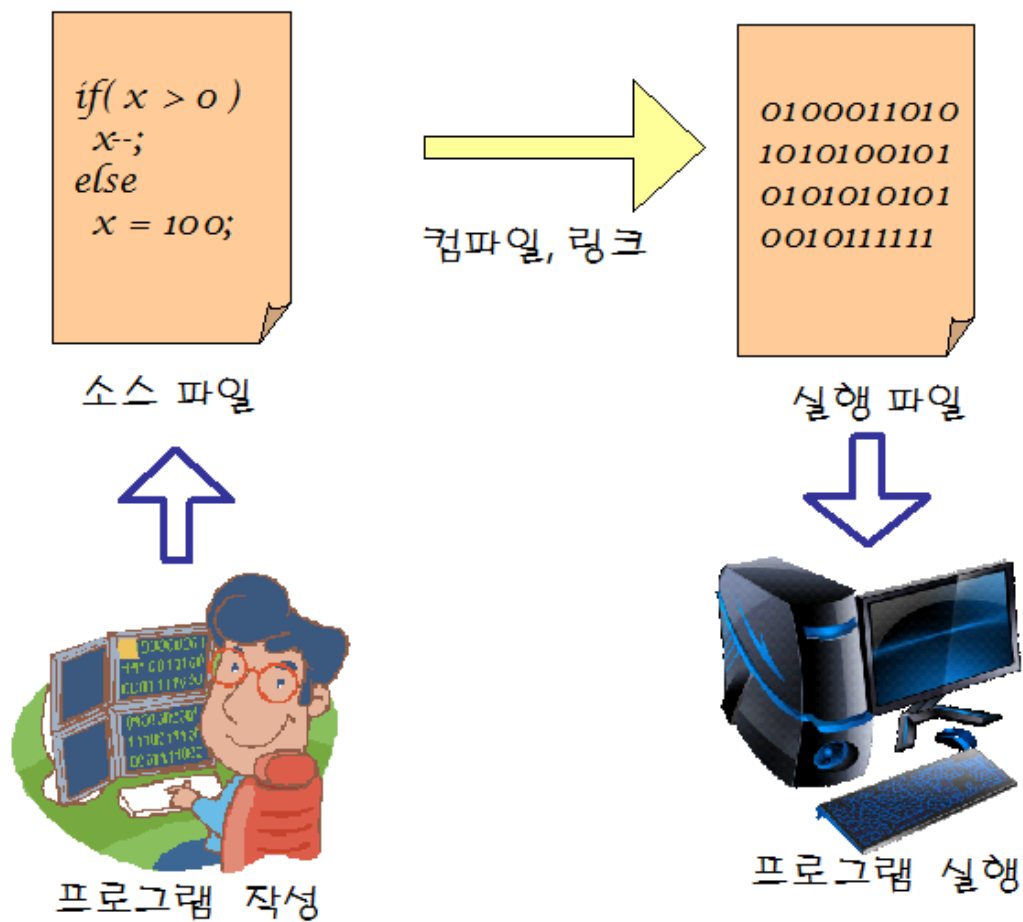


그림 1.15 프로그래밍의 순서

# 프로그램 작성 단계

- 편집 (*edit*)

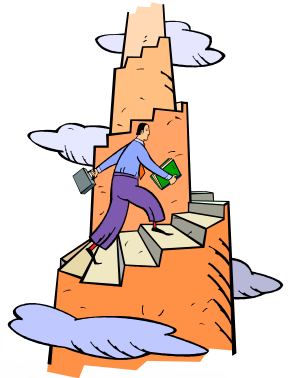
- 에디터를 이용하여 원하는 작업의 내용을 기술하여 소스 코드 작성
- 소스 파일(source file): 소스 코드가 들어 있는 텍스트 파일
  - (예) test.c

- 컴파일 (*compile*)

- 소스 파일->기계어로 변환
- 오브젝트 파일(object file) : 기계어로 변환된 파일
  - (예) test.obj

- 링크(*link*)

- 오브젝트 파일들을 라이브러리 파일들과 연결하여 하나의 실행 파일 생성
- 실행 파일 (executable file): 실행이 가능한 파일
  - (예) test.exe





# Q & A

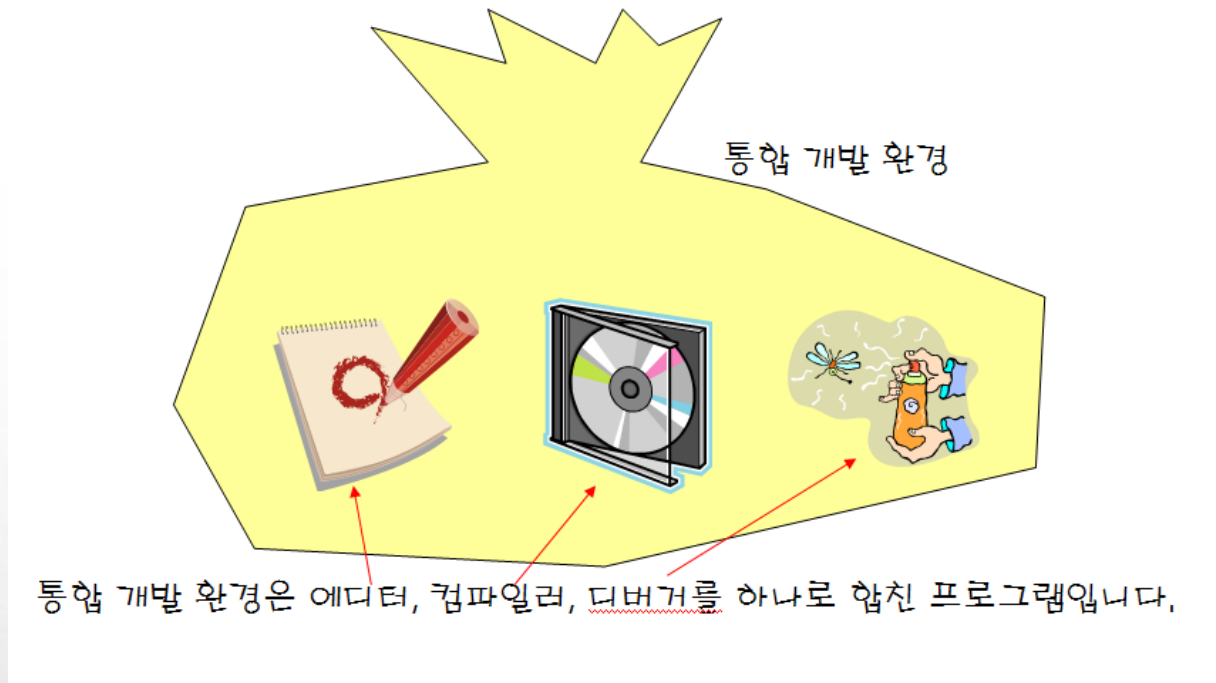


**Q 소스 파일과 오브젝트 파일, 실행 파일 중에서 반드시 보관하여야 하는 파일은 무엇일까?**

**A** 정답은 소스 파일이다. 소스 파일만 있으면 컴파일러를 수행시켜서 오브젝트 파일, 실행 파일은 만들 수 있다. 하지만 소스 파일을 삭제하면 컴파일이 불가능하다.

# 통합 개발 환경

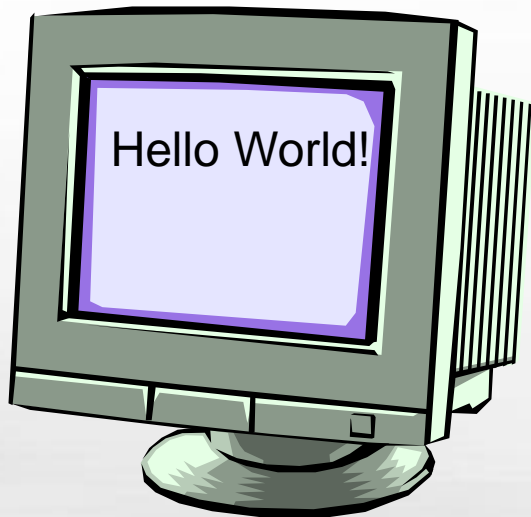
- 통합 개발 환경(IDE: integrated development environment):
  - 에디터 + 컴파일러 + 디버거



# 우리가 작성한 프로그램

```
/* 첫번째 프로그램*/  
#include <stdio.h>  
  
int main(void)  
{  
    printf("Hello World!");  
    return 0;  
}
```

다음 장에서  
자세히  
설명할  
것입니다.





# 중간 점검

1. 새로운 프로젝트를 생성하고 프로젝트에 소스 파일을 추가하는 메뉴는 무엇인가?
2. 프로젝트에 속하는 소스 파일을 컴파일하여 실행하는 메뉴는 무엇인가?
3. C 언어에서는 대문자와 소문자를 구별하는가?



# Q & A

