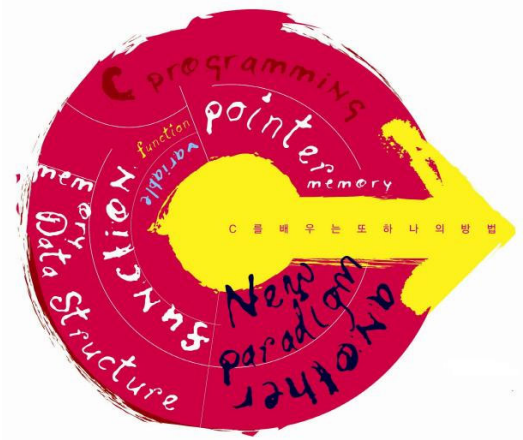
Clickseo.com

C Programming

전처리

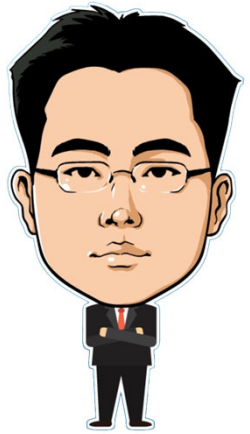
(Preprocessing)



Seo, Doo-Ok

Clickseo.com

clickseo@gmail.com



CC BY NC ND

목 차

CC BY NC ND

백문이불여일타(百聞而不如一打)

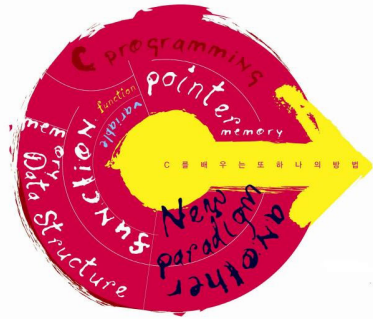
- 전처리의 이해
- 조건 및 분할 컴파일

전처리

CC BY NC ND

백문이불여일타(百聞而不如一打)

- 전처리의 이해
  - 매크로 상수
  - 매크로 함수
- 조건 및 분할 컴파일



전처리의 이해 (1/2)

- **전처리**(Preprocessing)
  - 원시 소스 파일을 컴파일 하기 전에 행해야 할 일련의 작업
    - 외부 파일 포함: **#include**
    - 매크로 상수: **#define**
    - 조건부 컴파일(Conditional Compile)
    - 시스템 내장 매크로
  - C 컴파일러
    - 전처리기(Preprocessor)와 번역기(Translator)로 구성

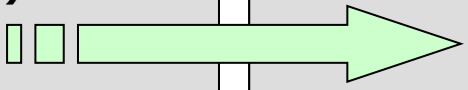
"전처리는 컴파일러가 동작하기 전 단계에서 작업이 이루어진다."

전처리의 이해 (2/2)

- **#define**

preprocessing
(macro expansion)

```
#define MAXSIZE    1024
#define ADD(a, b)  ((a) + (b))

int main(void)
{
        char    str[MAXSIZE];
        int     res;

        res = ADD(10, 20);

        return 0;
}
```



```
#define MAXSIZE    1024
#define ADD(a, b)  ((a) + (b))

int main(void)
{
        char    str[1024];
        int     res;

        res = ((10) + (20));

        return 0;
}
```

CC BY NC ND

전처리의 이해

매크로 상수

매크로 상수 (1/6)

● 매크로 상수

❍ 매크로 상수 정의

```
#define MAXSIZE     1024            // 정수형 상수
#define PI          3.14159         // 실수형 상수
#define TITLE       "Hi~ Clickseo"  // 문자열 상수

#define ESC         0x1b            // ESC: ASCII 코드 값
#define CR          0x0d            // CR: ASCII 코드 값
#define LF          0x0a            // LF: ASCII 코드 값

#define TRUE        1               // Boolean: true
#define FALSE       0               // Boolean: false
```

매크로 상수 (2/6)

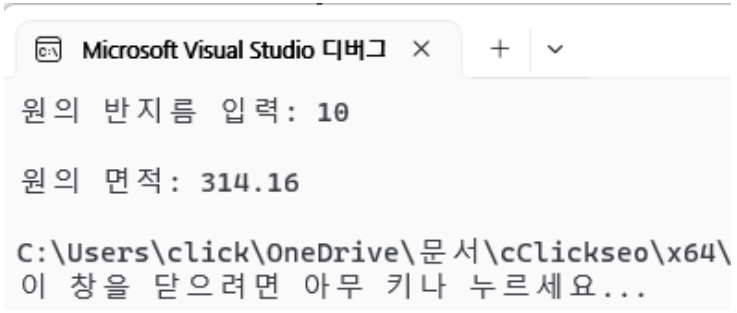
예제 10-1: 전처리의 이해 -- 매크로 상수

```c
#include <stdio.h>

#define PI 3.14159      // 매크로 상수

int main(void)
{
        double          area, radius;

        printf("원의 반지름 입력: ");
        scanf_s("%lf", &radius);
        // scanf("%lf", &radius);

        area = radius * radius * PI;

        printf("\n원의 면적: %.2f\n", area);
        return 0;
}
```

 # 전처리 사용 시...

프로그램 수정 용이

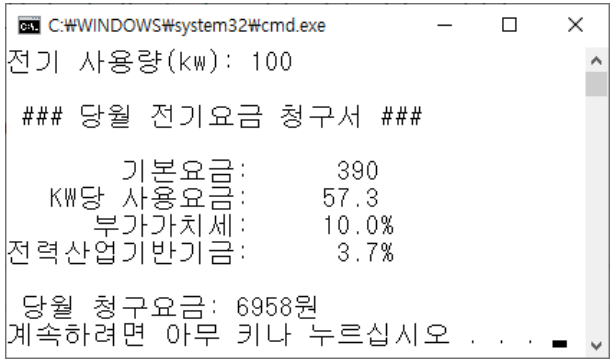
전체적인 프로그램 구조의 간결성 유지



매크로 상수 (3/6)

예제 10-2: 전기요금 계산 -- 매크로 상수 (1/2)

```
#include <stdio.h>

#define BASIC_RATE    390      // 기본요금
#define KW_RATE       57.3     // KW당 사용요금
#define MIN_CHARGE    1000     // 월 최저요금: 1,000
#define VAT           0.1      // 부가가치세: 10%
#define BASE_FUND     0.037    // 전력산업기반기금: 3.7%
#define TITLE         "당월 전기요금 청구서"

int main(void)
{
        int             fee, vatTax, fundTax, tot;
        double          kwh;

        printf("전기 사용량(kw): ");
        scanf_s("%lf", &kwh);                   // scanf("%lf", &kwh);
```



매크로 상수 (4/6)

예제 10-2: 전기요금 계산 -- 매크로 상수 (2/2)

```
        // 사용요금 = 기본료 + KW당 사용요금
        fee = (int)(BASIC_RATE + (kwh * KW_RATE));

        // 월 최저요금 적용: 1,000원
        if( fee < 1000 )
                fee = 1000;

        // 각종 세금: 부가가치세(10%), 전력산업기반기금(3.7%)
        vatTax = (int)(fee * VAT);
        fundTax = (int)(fee * BASE_FUND);

        // 전기료: 사용요금 + 부가가치세 + 전력산업기반기금
        tot = fee + vatTax + fundTax;

        printf("\n ### %s ###\n\n", TITLE);
        printf("        기본요금: %8d \n", BASIC_RATE);
        printf("  KW 당 사용요금: %8.1f \n", KW_RATE);
        printf("      부가가치세: %8.1f%%\n", VAT * 100);
        printf("전력산업기반기금: %8.1f%%\n", BASE_FUND * 100);

        printf("\n 당월 청구요금: %d원 \n", tot );

        return 0;
}
```

```
C:₩WINDOWS₩system32₩cmd.exe
전기 사용량(kw): 100

 ### 당월 전기요금 청구서 ###

        기본요금:      390
   KW당 사용요금:     57.3
      부가가치세:     10.0%
전력산업기반기금:      3.7%

 당월 청구요금: 6958원
계속하려면 아무 키나 누르십시오 . . .
```

매크로 상수 (5/6)

예제 10-3: 키보드 값 처리 -- 매크로 상수 (1/2)

```
#include <stdio.h>
#include <conio.h>    // _getch

// 기능 키
#define F1          59
#define F2          60
#define F3          61
#define F4          62
#define F5          63
#define F6          64
#define F7          65
#define F8          66
#define F9          67
#define F10         68

// 방향 키
#define UP          72
#define DOWN        80
#define LEFT        75
#define RIGHT       77

// 특수 키
#define ESC         27
#define ENTER       13
#define SPACE       32
#define BACKSPACE   8

#define INSERT      82
#define DELETE      83

#define HOME        71
#define END         79

#define PAGEUP      73
#define PAGEDOWN    81
```

```
Microsoft Visual Studio 디버그
입력된 키: 75    왼쪽 방향키
입력된 키: 77    오른쪽 방향키
입력된 키: 72    위쪽 방향키
입력된 키: 80    아래쪽 방향키

C:\Users\click\OneDrive\문서\cClickseo\x64\
이 창을 닫으려면 아무 키나 누르세요...
```

매크로 상수 (6/6)

예제 10-3: 키보드 값 처리 -- 매크로 상수 (2/2)

```
int main(void)
{
        int     ch;
        while( (ch = _getch()) != ESC)   {
                // 특수 키라면 두 번째 상위 바이트 값을 읽어온다.
                if(ch == 0 || ch == 224 || ch == -32)
                        ch = _getch();

                printf("입력된 키: %d \t", ch);
                switch(ch)      {
                        case LEFT:      printf("왼쪽 방향키 \n");       break;
                        case RIGHT:     printf("오른쪽 방향키 \n");     break;
                        case UP:        printf("위쪽 방향키 \n");       break;
                        case DOWN:      printf("아래쪽 방향키 \n");     break;
                        default: printf("나머지 키... \n");
                }
        }

        return 0;
}
```

```
Microsoft Visual Studio 디버그
입력된 키: 82   나머지 키...
입력된 키: 83   나머지 키...
입력된 키: 71   나머지 키...
입력된 키: 79   나머지 키...
입력된 키: 73   나머지 키...
입력된 키: 81   나머지 키...
입력된 키: 75   왼쪽 방향키
입력된 키: 77   오른쪽 방향키
입력된 키: 72   위쪽 방향키
입력된 키: 80   아래쪽 방향키
입력된 키: 59   나머지 키...
입력된 키: 60   나머지 키...
입력된 키: 61   나머지 키...
입력된 키: 62   나머지 키...
입력된 키: 63   나머지 키...
입력된 키: 64   나머지 키...
입력된 키: 65   나머지 키...
입력된 키: 66   나머지 키...
입력된 키: 67   나머지 키...
입력된 키: 68   나머지 키...

C:\Users\click\OneDrive\문서\cClickseo\x64\
이 창을 닫으려면 아무 키나 누르세요...
```

CC BY NC ND

전처리의 이해

매크로 함수

매크로 함수 (1/3)

- **매크로 함수**
  - 매크로 함수 정의

"함수 호출에 따르는 시간적 부담이 적다."

```
#define ADD(a, b)   ((a) + (b))
```

함수의 내용 함수의 몸체 정의(괄호 생략 주의)

인자 리스트
일반 함수의 형식 인자와 동일
단, 자료형을 명시하지는 않는다.
인자가 없을 수는 있지만 괄호를 생략할 수는 없다.

함수명
가급적 대문자로 작명하여
일반 함수와 구분해 주는 것이 좋다.

```
sum = ADD(10, 20);   ⟶   sum = (10 + 20);
```

preprocessing
(macro expansion)

매크로 함수 (2/3)

- **매크로 함수**
  - 매크로에 의한 치환은 문자열 내에서는 이루어지지 않는다.

```
#include <stdio.h>
#define ADD(a, b) printf("a + b = %d₩n", a + b )

int main(void)
{
    ADD(10, 20);
    return 0;
}
```

```
Microsoft Visual Studio 디버그
a + b = 30

C:\Users\click\OneDrive\문서\cClickseo\x64\
이 창을 닫으려면 아무 키나 누르세요...
```

```
#include <stdio.h>
#define ADD(a, b) printf(#a " + " #b " = %d₩n", a + b )

int main(void)
{
    ADD(10, 20);
    return 0;
}
```

```
#define OUTPUT(exp)  printf(#exp " = %d", exp)
```

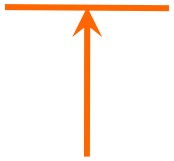
```
Microsoft Visual Studio 디버그
10 + 20 = 30

C:\Users\click\OneDrive\문서\cClickseo\x64\
이 창을 닫으려면 아무 키나 누르세요...
```

매크로 함수 (3/3)

- **매크로 함수**
  - ## : 토큰(token)을 결합할 때 사용
    - token: 컴파일러가 인식하는 의미를 지니는 문자나 문자열의 최소 단위

      (변수나 함수의 이름을 동적으로 작성하기 위하여 사용)

```
#define CAT(a, b)  a##b
```



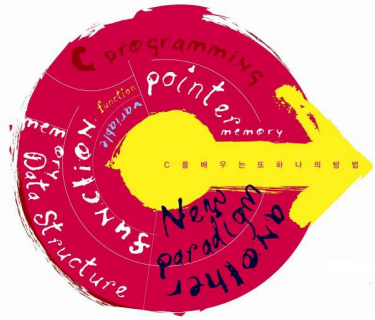
res = CAT(n, Value); 문장은 res = nValue; 의 문장으로 처리된다.

조건 및 분할 컴파일

CC BY NC ND

백문이불여일타(百聞而不如一打)

- 전처리의 이해
- **조건 및 분할 컴파일**
  - 조건 컴파일
  - 시스템 내장 매크로
  - 분할 컴파일



조건 컴파일 (1/5)

- **조건 컴파일**(conditional compile)
  - 특정 조건을 만족할 경우에 지정한 범위 내의 문장을 컴파일한다.

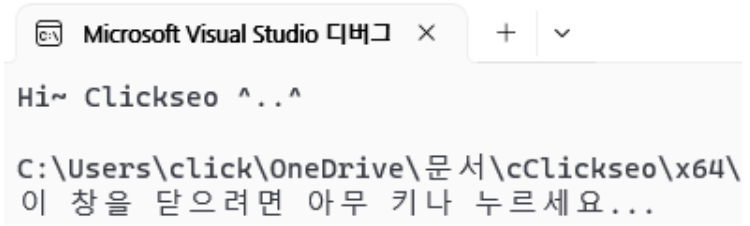
  - 조건 컴파일 지시자 사용 형식
    - { } 를 사용하지 않기 때문에 #endif 생략불가
    - ++, --, &, * 와 기타 복합 연산자 사용 제한
    - #ifdef 와 #ifndef 는 #if defined(매크로명), #if !defined(매크로명)

#if 상수식
  실행문장영역#1
#elif 상수식
  실행문장영역#2
#else
  실행문장영역#N
#endif
	#ifdef 매크로명
	  실행문장영역#1
	#elif 매크로명
	  실행문장영역#2
	#else
	  실행문장영역#N
	#endif
		#ifndef 매크로명
		  실행문장영역#1
		#elif 매크로명
		  실행문장영역#2
		#else
		  실행문장영역#N
		#endif

조건 컴파일 (2/5)

예제 10-4: 조건 컴파일 -- #if ~ #elif ~ #endif

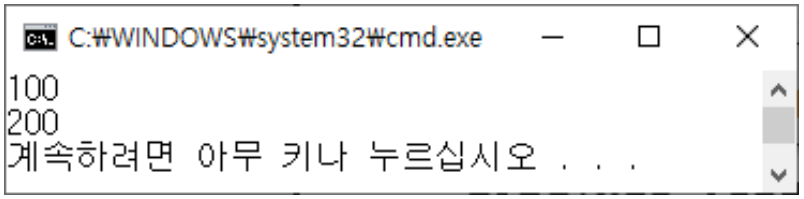
```
#include <stdio.h>
#if 0
int main(void)
{
        printf("Hello World!!! \n");
        return 0;
}
#elif 1
int main(void)
{
        printf("Hi~ Clickseo ^..^ \n");
        return 0;
}
#endif
```



조건 컴파일 (3/5)

예제 10-5: 조건 컴파일 -- #ifdef, #undef

```
#include <stdio.h>

#define  MAXSIZE         1024
#define  OUTPUT(num)     printf("%d₩n", num);

int main(void)
{
        int a, b;

        a = MAXSIZE;
        OUTPUT(a);

#ifdef   MAXSIZE
#undef   MAXSIZE
#endif
#define MAXSIZE 256

        b = MAXSIZE;
        OUTPUT(b);

        return 0;
}
```




```
// #undef : 이미 정의된 매크로 기능을 없던 일로 되돌린다.


#ifdef  TITLE
   #undef  TITLE
#endif
#define TITLE "COMPUTER SCIENCE"
```

조건 컴파일 (4/5)

● 조건 컴파일

❍ #error

- 오류가 발생할 경우 오류 메시지를 출력하기 위한 용도로 사용
- #error 문을 만나면 컴파일을 즉시 중단하고, 시스템에서 미리 정의된 오류 메시지의 일부로 errmsg 를 포함하여 출력한다.

```
#if REQUIRED_MEMORY > 1024
      ...
#else
   #error "OOPs! This program can be used with 1024 RAM... "
#endif
```

❍ #line

- 현재 컴파일 되고 있는 프로그램의 행 번호와 파일명을 변경하고자 할 때 사용

```
// 다음 라인을 1001번 라인으로 변경하고, 파일명은 Clickseo.c 로 설정한다.
#line 1000 "Clickseo.c"
#error "see this error message..."
      ... 에러가 자주 발생하는 영역 ...
```

조건 컴파일 (5/5)

- **조건 컴파일**
  - **#pragma**
    - 컴파일러에게 특수한 기능이나 동작을 수행하게끔 지시하는 용도

```
// Function should return a value 경고를 발생하지 못하게 한다.
#pragma warn –rvl

// long 형 상수에 접미사 L을 붙이지 않으면 경고를 발생케 한다.
#pragma warn +cln
```

❖ 주의: 컴파일러의 종류에 따라 용법에 차이가 있다.

CC BY NC ND

조건 및 분할 컴파일

시스템 내장 매크로

시스템 내장 매크로 (1/3)

● 시스템 내장 매크로

❍ 시스템에서 제공하는 미리 내장된 매크로

내장 매크로	기 능
__STDC__	컴파일러의 ANSI C 표준 준수 여부(1이면 준수)
__FILE__	현재 컴파일 되고 있는 파일의 이름
__LINE__	현재 컴파일 되고 있는 프로그램의 행 번호
__DATE__	컴파일 날짜(월,일,년)
__TIME__	컴파일이 시작된 시간(시:분:초)
__TIMESTAMP__	컴파일이 시작된 날짜와 시간

시스템 내장 매크로 (2/3)

예제 10-6: 시스템 내장 매크로 -- 컴파일 정보 변경

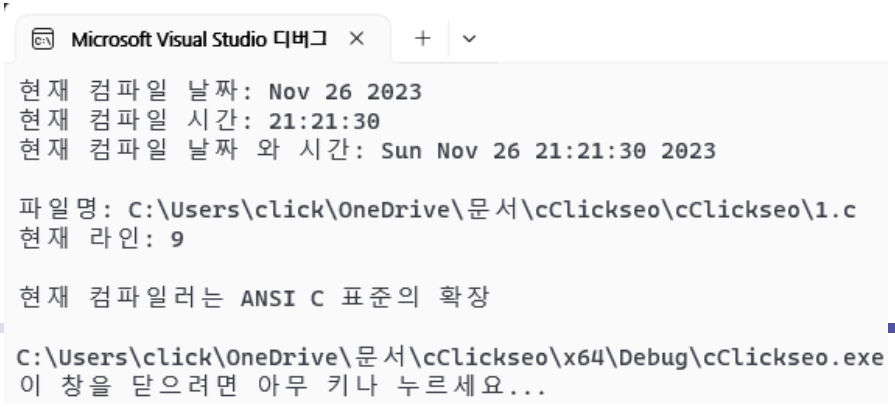
```
#include <stdio.h>
int main(void)
{
    printf("컴파일이 시작된 날짜와 시간: %s \n", __TIMESTAMP__ );
    printf("컴파일 관련 정보: %s, line: %d, on %s, at %s\n",
                                __FILE__, __LINE__, __DATE__, __TIME__ );

#line 1000 "Clickseo.c"

    printf("변경된 컴파일 정보 \n");
    printf("컴파일 관련 정보: %s, line: %d, on %s, at %s\n",
                                __FILE__, __LINE__, __DATE__, __TIME__ );
    return 0;
}
```

```
Microsoft Visual Studio 디버그
컴파일이 시작된 날짜와 시간: Sun Nov 26 21:14:28 2023
컴파일 관련 정보: C:\Users\click\OneDrive\문서\cClickseo\cClickseo\1.c, line: 6, on Nov 26 2023, at 21:14:28
변경된 컴파일 정보
컴파일 관련 정보: Clickseo.c, line: 1003, on Nov 26 2023, at 21:14:28

C:\Users\click\OneDrive\문서\cClickseo\x64\Debug\cClickseo.exe(프로세스 8968개)이(가) 종료되었습니다(코드: 0개).
이 창을 닫으려면 아무 키나 누르세요...
```

시스템 내장 매크로 (3/3)

예제 10-7: 시스템 내장 매크로 -- ANSI 표준과 확장

```
#include <stdio.h>
int main(void)
{
        printf("현재 컴파일 날짜: %s\n", __DATE__ );
        printf("현재 컴파일 시간: %s\n", __TIME__ );
        printf("현재 컴파일 날짜 와 시간: %s\n\n", __TIMESTAMP__ );

        printf("파일명: %s\n", __FILE__ );
        printf("현재 라인: %d\n\n", __LINE__ );

#ifdef __STDC__
        printf("현재 컴파일러는 ANSI 표준 \n");
#else
        printf("현재 컴파일러는 ANSI C 표준의 확장 \n");
#endif

        return 0;
}
```



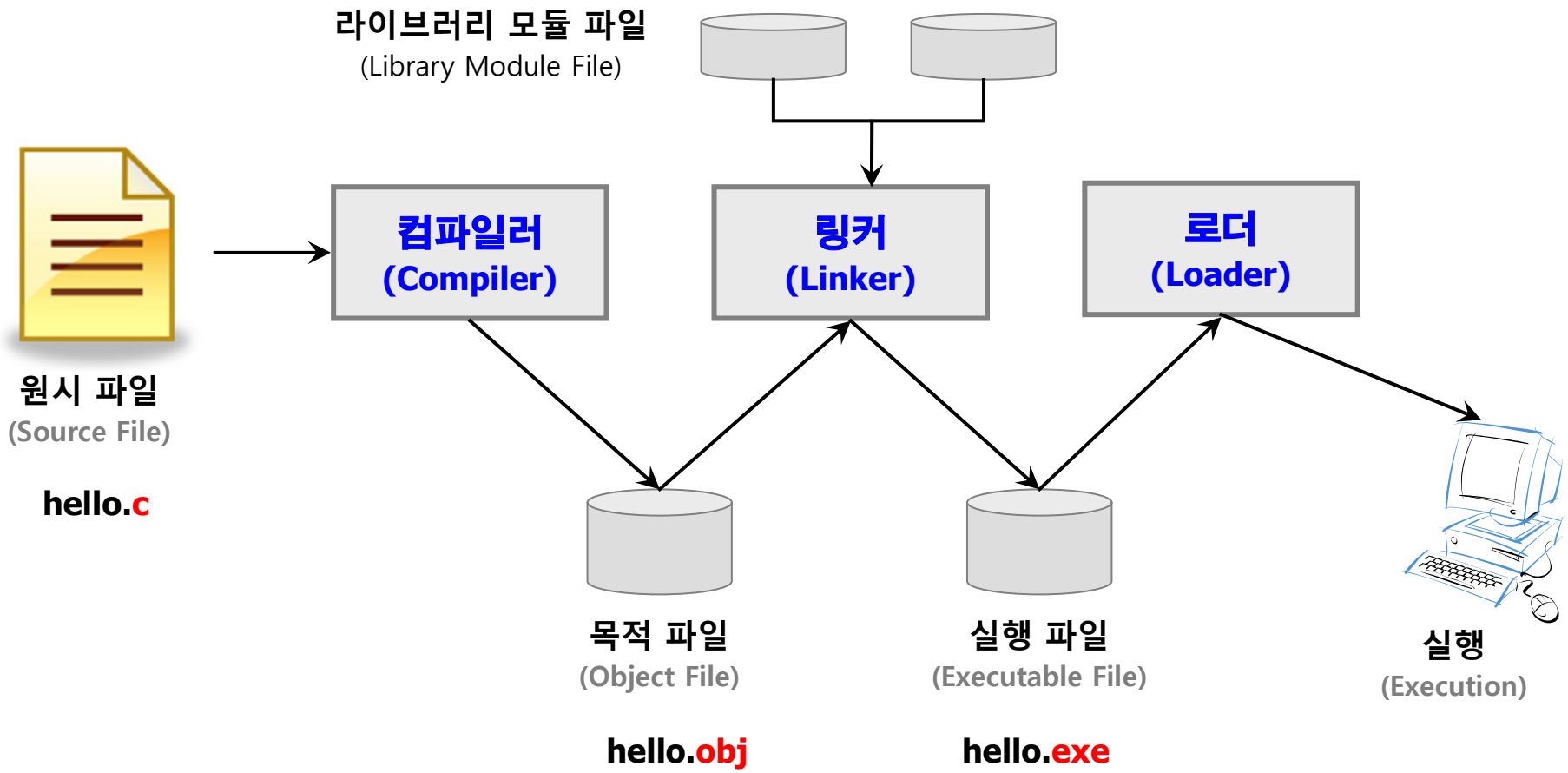
CC BY NC ND

조건 및 분할 컴파일

분할 컴파일

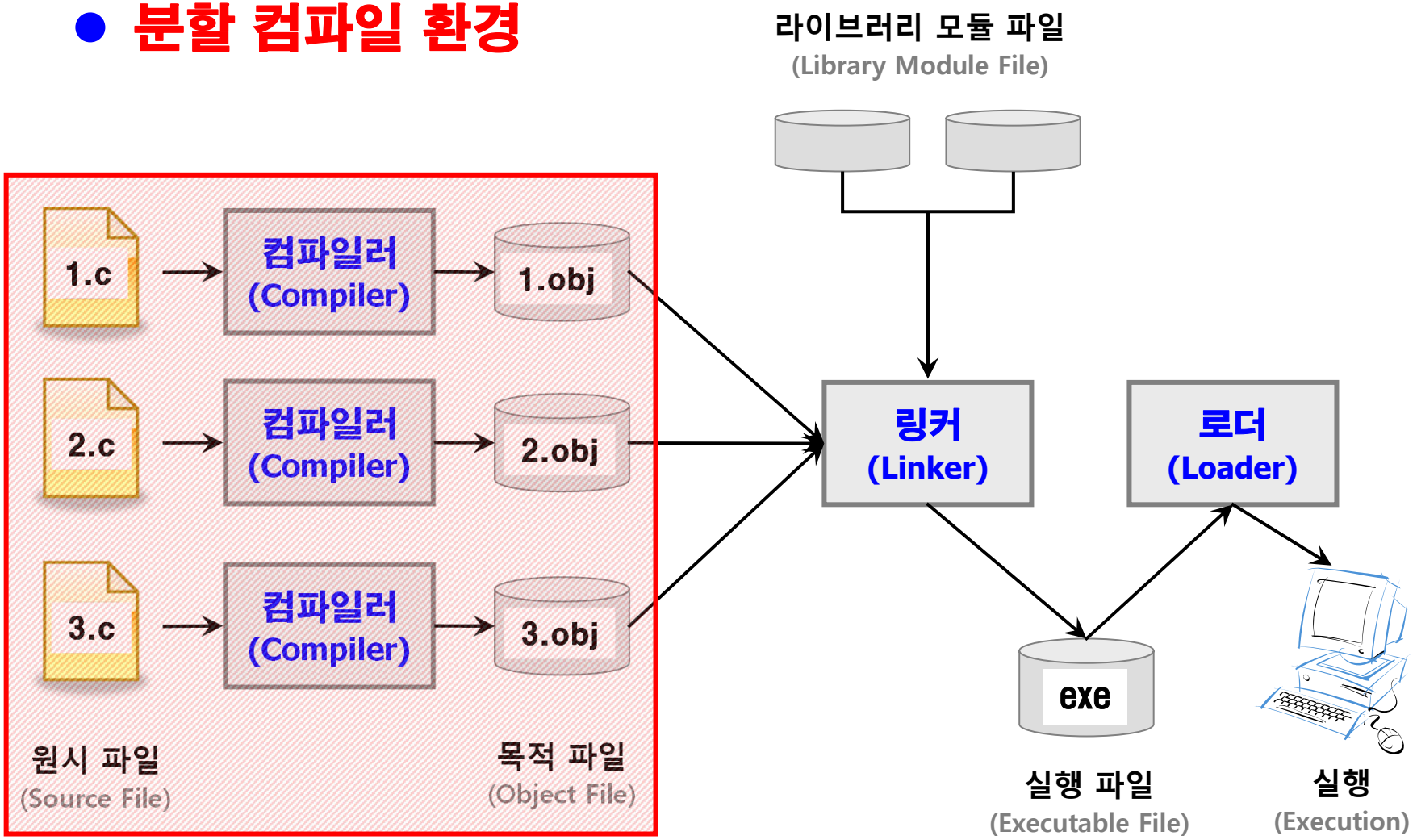
분할 컴파일 (1/7)

- 단일 컴파일 환경



분할 컴파일 (2/7)

- **분할 컴파일 환경**



분할 컴파일 (3/7)

- **#include**
  - 외부의 헤더 파일(.h)에 정의된 내용을 현재 프로그램의 내부로 포함시키는 역할

```
#include <시스템 정의 파일명>
#include "사용자 정의 파일명"
```

```
#include <stdio.h>
#include <stdlib.h>
#include "test.h"
// 절대 경로의 사용은 좋은 습관이 아니다.
#include  C:\clickseo\test.h"                          // Windows
#include "/home/clickseo/my/include/test.h"            // UNIX/Linux
```

score.h

```
// #pragma once
#ifndef arrMAXSIZE
        #define arrMAXSIZE 3
#endif
#ifndef __SCORE_H_
#define __SCORE_H_
typedef  struct  _score {
         char     name[12];
         int      kor, eng, math, tot;
         float    ave;
} SCORE;
#endif

void     INPUT(SCORE *pArr);
void     OUTPUT(SCORE *pArr);
```

main.c

```
#include "score.h"

int main(void)
{
        ...
}
```

input.c

```
#include <stdio.h>
#include "score.h"

void  INPUT(SCORE *pArr)
{

}
```

output.c

```
#include <stdio.h>
#include "score.h"

void  OUTPUT(SCORE *pArr)
{

}
```

main.obj + input.obj + output.obj → score.exe

분할 컴파일 (5/7)

예제 10-8: 성적 관리 프로그램 -- 분할 컴파일 score.h

```
// #pragma once
#ifndef arrMAXSIZE
        #define arrMAXSIZE 3
#endif
#ifndef __SCORE_H_
#define __SCORE_H_
typedef struct _score {
        char    name[12];
        int     kor, eng, math, tot;
        float   ave;
} SCORE;
#endif

void    input(SCORE *pArr, int num);
void    print(SCORE *pArr, int num);
```

예제 10-8: 성적 관리 프로그램 -- 분할 컴파일 main.c

```
#include "score.h"
int main(void)
{
        SCORE   arr[arrMAXSIZE];
        input(arr, arrMAXSIZE);
        print(arr, arrMAXSIZE);
        return 0;
}
```

분할 컴파일 (6/7)

예제 10-8: 성적 관리 프로그램 -- 분할 컴파일 input.c

```
#include <stdio.h>
#include "score.h"

void  input(SCORE *pArr, int num) {
        for(int i=0; i<num; ++i)      {
                printf("\n %d 번째 학생 성적 입력... \n", i+1);
                printf("학생 이름: ");
                gets_s(pArr[i].name, sizeof(pArr[i].name));
                // gets(pArr ->name);

                printf("국어 점수: ");
                scanf_s("%d", &pArr[i].kor); // scanf("%d", &pArr[i].kor);
                printf("영어 점수: ");
                scanf_s("%d", &pArr[i].eng); // scanf("%d", &pArr[i].eng);
                printf("수학 점수: ");
                scanf_s("%d%*c", &pArr[i].math);
                // scanf_s("%d%*c", & pArr[i].math);

                pArr[i].tot = pArr[i].kor + pArr[i].eng + pArr[i].math;
                pArr[i].ave = (float)pArr[i].tot / 3;
        }
}
```

분할 컴파일 (7/7)

예제 10-8: 성적 관리 프로그램 -- 분할 컴파일 output.c

```
#include <stdio.h>
#include "score.h"

void  print(SCORE *pArr, int num) {
        printf("\n \t ### 학생 성적 ### \n");
        for(int i=0; i<num; ++i, ++pArr)
                printf("%10s %3d %3d %3d %5d %8.2f \n",
                        pArr->name, pArr->kor, pArr->eng, pArr->math,
                        pArr->tot, pArr->ave);
}
```

```
Microsoft Visual Studio 디버그

 1 번째 학생 성적 입력...
학생 이름: 홍길동
국어 점수: 75
영어 점수: 85
수학 점수: 95

 2 번째 학생 성적 입력...
학생 이름: 이순신
국어 점수: 77
영어 점수: 88
수학 점수: 99

 3 번째 학생 성적 입력...
학생 이름: 이도
국어 점수: 88
영어 점수: 89
수학 점수: 98

        ### 학생 성적 ###
    홍길동  75  85  95   255    85.00
    이순신  77  88  99   264    88.00
      이도  88  89  98   275    91.67

C:\Users\click\OneDrive\문서\cClickseo\x64\
이 창을 닫으려면 아무 키나 누르세요...
```

참고문헌

[1] 서현우, “혼자 공부하는 C 언어: 1:1 과외 하듯 배우는 프로그래밍 자습서”, 한빛미디어, 2023.

[2] Paul Deitel, Harvey Deitel, "C How to Program", Global Edition, 8/E, Pearson, 2016.

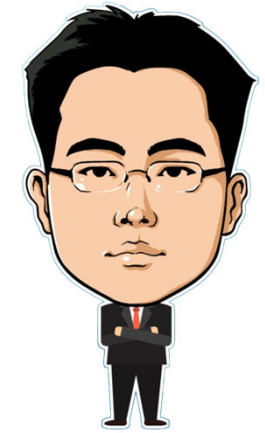
[3] Kamran Amini, 박지윤 번역, “전문가를 위한 C : 동시성, OOP부터 최신 C, 고급 기능까지!”, 한빛미디어, 2022.

[4] 서두옥, “(열혈강의) 또 하나의 C : 프로그래밍은 셀프입니다”, 프리렉, 2012.

[5] Behrouz A. Forouzan, Richard F. Gilberg, 김진 외 7인 공역, “구조적 프로그래밍 기법을 위한 C”, 도서출판 인터비젼, 2004.

[6] Brian W. Kernighan, Dennis M. Ritchie, 김석환 외 2인 공역, “The C Programming Language”, 2/E, 대영사, 2004.

[7] "C reference", cppreference.com, 2024 of viewing the site, https://en.cppreference.com/w/c.



이 강의자료는 저작권법에 따라 보호받는 저작물이므로 무단 전제와 무단 복제를 금지하며,
내용의 전부 또는 일부를 이용하려면 반드시 저작권자의 서면 동의를 받아야 합니다.

Copyright © Clickseo.com. All rights reserved.