

사칙연산 프로그램

제주대학교 컴퓨터공학과

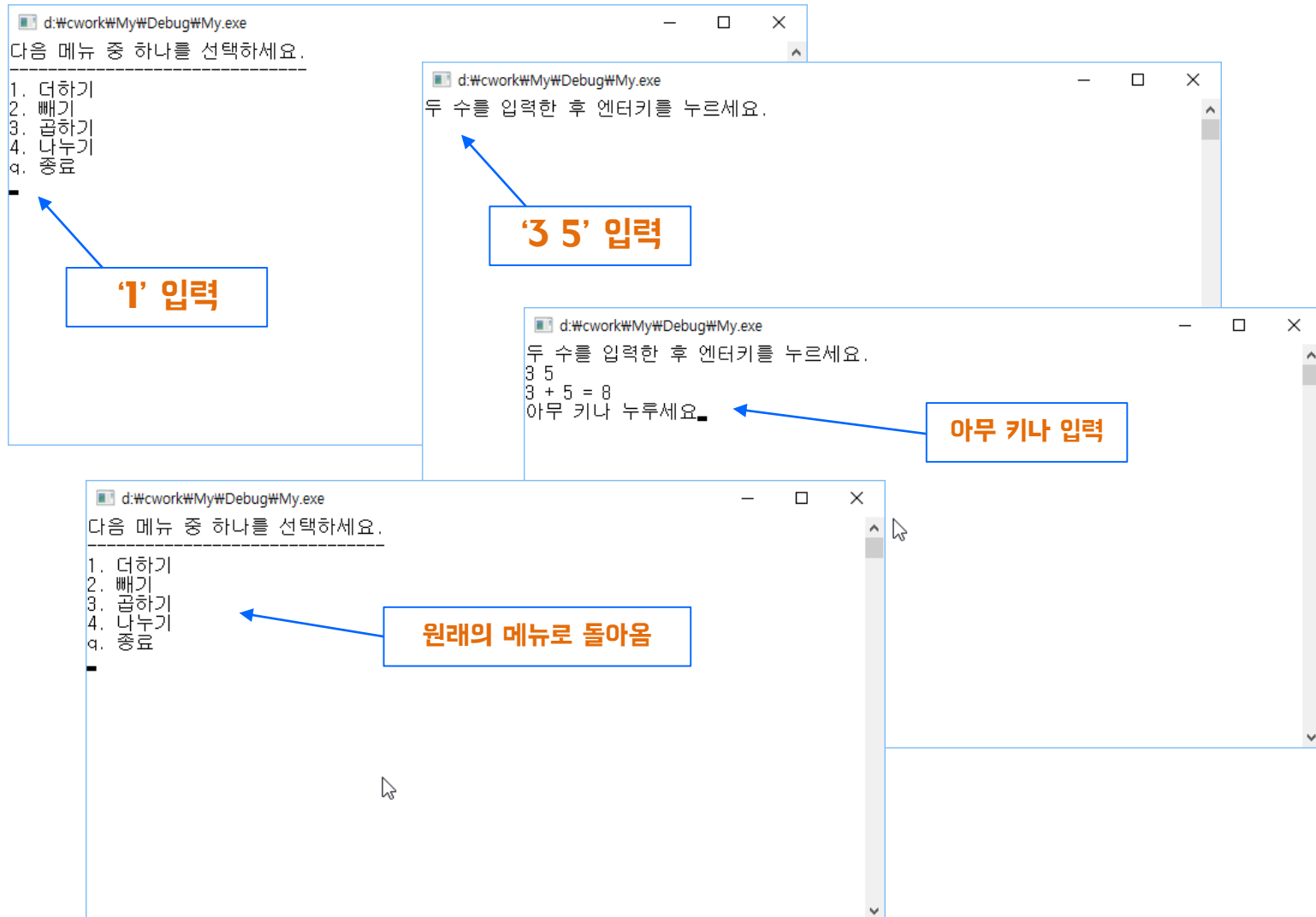
변영철 교수

(ycb@jejunu.ac.kr)

이 장을 공부하면

- 사칙연산 프로그램을 작성할 수 있다.
- 코드 추상화에 익숙해진다.
- 라이브러리 모듈을 작성할 수 있다.
- 내가 작성한 라이브러리를 이용하여 프로그램을 쉽게 작성할 수 있다.

사칙연산 프로그램 실행 화면



프로젝트 My 작성

- 파일 | 새로 만들기 | 프로젝트
- Visual C++ 선택 후 '빈 프로젝트' 선택
- 정보 입력
 - 이름 : My
 - 위치 :
- 확인 버튼 클릭
- 프로젝트 | 새 항목 추가
 - My.cpp 파일 추가

메뉴 표시하기

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

void main()
{
    while (1) {
        printf("다음 메뉴 중 하나를 선택하세요.\n");
        printf("-----\n");
        printf("1. 더하기\n");
        printf("2. 빼기\n");
        printf("3. 곱하기\n");
        printf("4. 나누기\n");
        printf("q. 종료\n");

        _getch();
    }
}
```

코드 추상화 - DisplayMenu

```
void DisplayMenu() {  
    printf("다음 메뉴 중 하나를 선택하세요.\n");  
    printf("-----\n");  
    printf("1. 더하기\n");  
    printf("2. 빼기\n");  
    printf("3. 곱하기\n");  
    printf("4. 나누기\n");  
    printf("q. 종료\n");  
}
```

```
void main() {  
    while (1) {  
        DisplayMenu();  
  
        _getch();  
    }  
}
```

프로그램 종료 구현

- 입력한 문자가 'q'이면 종료(exit)

```
char ch = _getch();  
switch (ch) {  
case 'q':  
    exit(1);  
}  
}  
}
```

메뉴 선택과 switch/case 문

```
char ch = _getch();
switch (ch) {
    case '1':
        break;
    case '2':
        break;
    case '3':
        break;
    case '4':
        break;
    case 'q':
        exit(1);
}
}
```


더하기 구현

- 두 수를 입력 받아 **더하여 출력**

```
char ch = _getch();  
switch (ch) {  
case '1':  
    int a, b;  
    scanf_s("%d %d", &a, &b);  
    printf("%d + %d = %d\n", a, b, a + b);  
    break;
```

&a: 지역변수 a가 만들어진 곳



배기 구현

case '2':

```
scanf_s("%d %d", &a, &b);  
printf("%d - %d = %d\n", a, b, a - b);  
break;
```

case '3':

곱하기 구현

case '3':

```
scanf_s("%d %d", &a, &b);  
printf("%d * %d = %d\n", a, b, a * b);  
break;
```

case '4':

나누기 구현

case '4':

```
scanf_s("%d %d", &a, &b);  
printf("%d / %d = %f\n", a, b, float(a) / b);  
break;
```

case 'q':

캐스팅

- 정수 / 정수 → 모두가 정수면 **결과는 정수**
- 실수 / 정수 → **어느 하나라도 실수이면** 결과는 실수
- 따라서 정수 변수 a, 혹은 b를 실수로 만들어야 하며(자료 형 변환), 이를 **캐스팅**이라고 함.

화면 정리 기능 구현

- 메뉴 1, 2, 3, 4 중 하나를 선택할 경우 화면 clear 및 안내 메시지 표시

```
char ch = _getch();  
switch (ch) {  
case '1':
```

```
    system("cls");  
    printf("두 수를 입력한 후 엔터키를 누르세요.\n");
```

```
    int a, b;  
    scanf_s("%d %d", &a, &b);  
    printf("%d + %d = %d\n", a, b, a + b);
```

화면 정리 기능 구현

- 결과를 표시한 후 안내 메시지 표시 및 멈춤

```
case '1':  
    system("cls");  
    printf("두 수를 입력한 후 엔터키를 누르세요.\n");
```

```
    int a, b;  
    scanf_s("%d %d", &a, &b);  
    printf("%d + %d = %d\n", a, b, a + b);
```

```
    printf("아무 키나 누르세요");  
    _getch();  
    system("cls");
```

```
    break;
```

코드 추상화

```
void main()
{
    while (1) {
        DisplayMenu();

        char ch = _getch();
        switch (ch) {
            case '1':
                Before();
                Plus();
                After();
                break;
            case '2':
```

코드 추상화

```
void Before() {  
    system("cls");  
    printf("두 수를 입력한 후 엔터키를 누르세요.\n");  
}  
  
void After() {  
    printf("아무 키나 누르세요");  
    _getch();  
    system("cls");  
}  
  
void Plus() {  
    int a, b;  
    scanf_s("%d %d", &a, &b);  
    printf("%d + %d = %d\n", a, b, a + b);  
}
```


코드 추상화

- 오류 수정

- 변수 a, b를 Plus 함수로 옮겼으므로 이를 사용하는 다른 곳에서 변수가 없다는 오류가 발생
- 변수를 정의하면 오류가 해결

```
case '2':
```

```
    int a, b;
```

```
    scanf_s("%d %d", &a, &b);
```

```
    printf("%d - %d = %d\n", a, b, a - b);
```

```
    break;
```

```
case '3':
```

메뉴 선택 시 추가 기능 구현

- 뽕셈 구현
- 곱셈 구현
- 나눗셈 구현

메뉴 선택 시 추가 기능 구현

```
case '2':  
    Before();  
    Minus();  
    After();
```

```
    break;
```

```
case '3':  
    Before();  
    Multiply();  
    After();
```

```
    break;
```

```
case '4':  
    Before();  
    Divide();  
    After();
```

```
    break;
```

```
case 'q':  
    exit(1);
```

```
}
```

인트로 화면

```
void main()
{
    printf("계산기 프로그램 프로그램에 오신 걸 환영합니다!");
    Sleep(3000);
    system("cls");

    while (1) {
        DisplayMenu();
    }
}
```

코드 추상화

- 인트로 코드

```
void ShowIntro() {  
    printf("계산기 프로그램 프로그램에 오신 걸 환영합니다!");  
    Sleep(3000);  
    system("cls");  
}
```

```
void main()  
{
```

```
    ShowIntro();
```

```
    while (1) {  
        DisplayMenu();
```

코드 추상화

```
void GetUserInputAndProcess() {  
    char ch = _getch();  
    switch (ch) {  
        case '1':  
            //... (생략)  
        case 'q':  
            exit(1);  
    }  
}
```

```
void main()  
{  
    ShowIntro();  
  
    while (1) {  
        DisplayMenu();  
        GetUserInputAndProcess();  
    }  
}
```

모듈별 분할 컴파일 - cpp 파일 작성

- 프로젝트 | 새 항목 추가 선택
- C++ 파일 선택한 후 Calculator.cpp 파일 추가
- 함수들을 이곳으로 이동

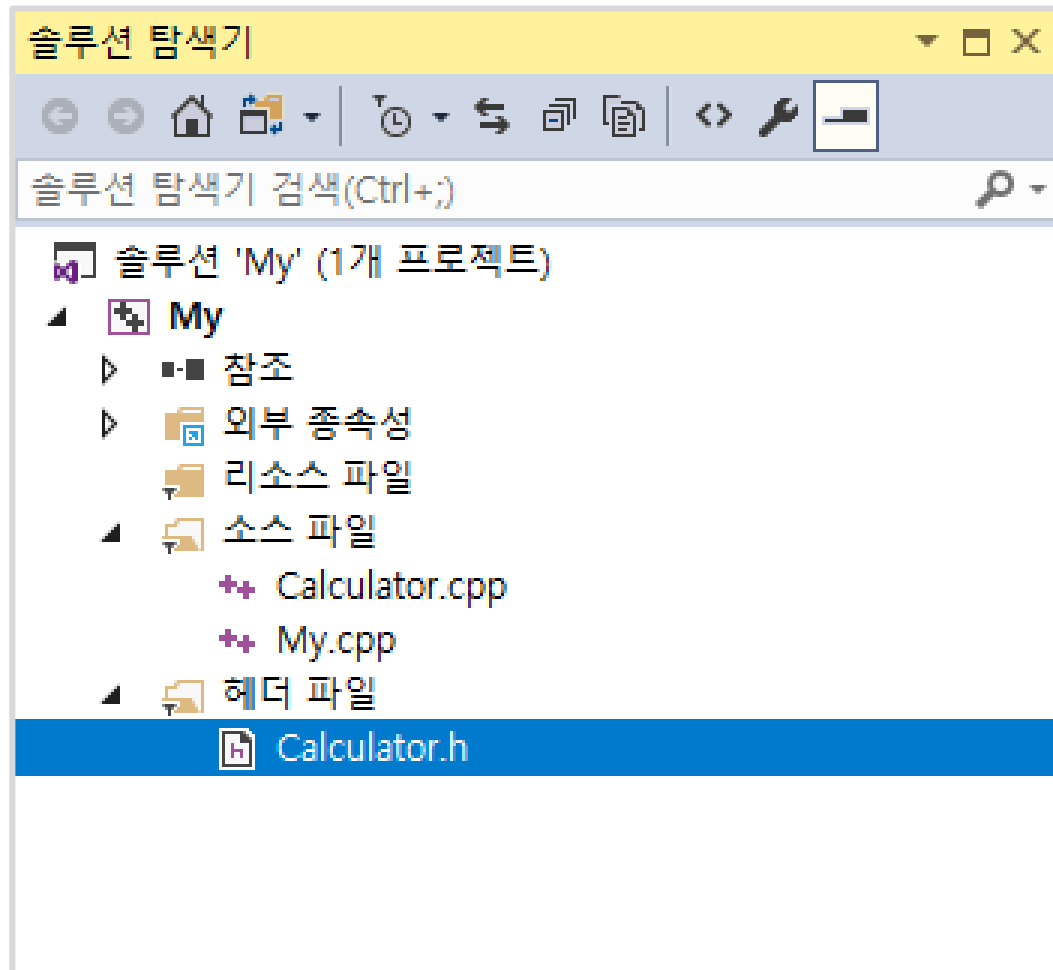
모듈별 분할 컴파일 - 헤더 파일 작성

- 프로젝트 | 새 항목 추가 선택
- 헤더 파일 선택한 후 Calculator.h 파일 추가
- 다음과 같이 함수 선언 작성

#pragma once

```
void DisplayMenu();  
void Before();  
void After();  
void Plus();  
void Minus();  
void Multiply();  
void Divide();  
void ShowIntro();  
void GetUserInputAndProcess();
```


모듈별 분할 컴파일 - 헤더 파일 작성



오류 수정

- My.cpp에서 발생하는 오류 수정

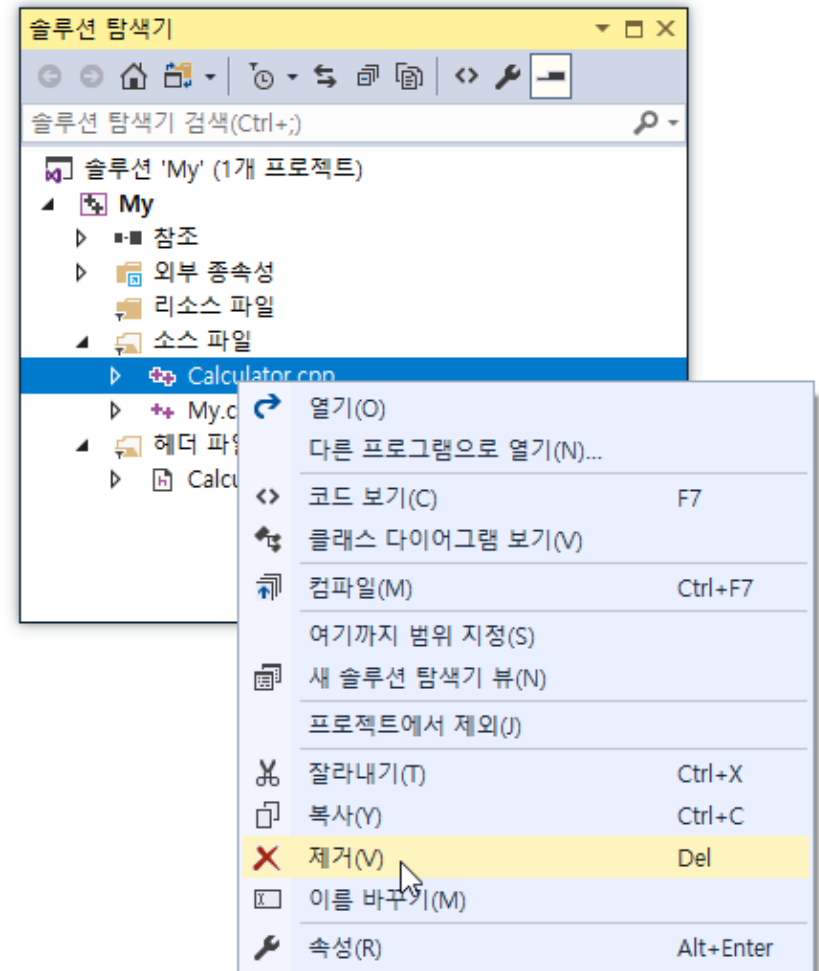
```
#include "Calculator.h"
```

```
void main()
{
    ShowIntro();

    while (1) {
        DisplayMenu();
        GetUserInputAndProcess();
    }
}
```

라이브러리 폴더로 옮기기

- Calculator.cpp, Calculator.h 두 파일을 clib 폴더로 이동
- 두 파일을 프로젝트에서 제거



라이브러리 이용하기

- 라이브러리 **파일 추가**
 - 프로젝트 | 기존 항목 추가...
 - c:\Wclib 폴더로 이동하여 Hello.cpp 추가
- 헤더 파일 **경로 추가**
 - 프로젝트 | 속성 선택
 - C/C++ | 일반 선택
 - 오른쪽 맨 위의 '추가 포함(include) 디렉터리'에 라이브러리 폴더 입력 - c:\Wclib
- 컴파일 및 실행

라이브러리를 이용한 새로운 프로그램

1. Visual C++ 빈 프로젝트(CalcApp) 생성
2. 헤더 파일 **경로 추가**
3. 새 항목(My.cpp) 추가 및 코드 작성

```
#include <Calculator.h>

void main() {
    ShowIntro();
    while (1) {
        DisplayMenu();
        GetUserInputAndProcess();
    }
}
```

라이브러리를 이용한 새로운 프로그램

- 라이브러리 파일 추가
 - 프로젝트에 기존 항목 추가
- 컴파일 및 실행

요약

- 사칙연산 프로그램을 작성할 수 있다.
- 코드 추상화에 익숙해진다.
- 라이브러리 모듈을 작성할 수 있다.
- 내가 작성한 라이브러리를 이용하여 프로그램을 쉽게 작성할 수 있다.