

소스 파일 구현 설명

202304148 조하린

▶ 다수의 클래스를 선언하고 활용하는 간단한 문제이다. 더하기(+), 빼기(-), 곱하기(*), 나누기(/)를 수행하는 4개의 클래스 Add, Sub, Mul, Div를 만들고자 한다. 이들은 모두 공통으로 다음 멤버를 가진다.

- int 타입 변수 a, b : 피연산자
- void setValue(int x, int y) 함수 : 매개 변수 x, y를 멤버 a, b에 복사
- int calculate() 함수 : 연산을 실행하고 결과 리턴

main() 함수는 Add, Sub, Mul, Div 클래스 타입의 객체를 a, s, m, d를 생성하고, 키보드로부터 두 개의 정수와 연산자를 입력받고, a, s, m, d 객체 중에서 연산을 처리할 객체의 setValue() 함수를 호출한 후, calculate()를 호출하여 결과를 화면에 출력한다. 프로그램은 무한 루프를 돈다.

문제 정의

더하기(+), 빼기(-), 곱하기(*), 나누기(/)를 수행하는 4개의 클래스 Add, Sub, Mul, Div를 만들고 키보드로부터 두 개의 정수와 연산자를 입력받아 연산 결과를 화면에 출력한다.

1. 클래스 : Add, Sub, Mul, Div 4개의 클래스는 각각에 대해 int 타입 변수 a, b를 멤버로 가지고, setValue() 함수와 calculate() 함수를 구현한다.
2. 입력 : 키보드로부터 두 개의 정수와 연산자를 입력 받고, 연산자 자리에 4개의 연산자 이외에 것들을 입력했을 경우, 다시 입력해달라는 메시지를 출력하고 다시 입력받는다.
3. 무한 루프 : 프로그램은 무한 루프를 돌아 입력을 계속 받는다.

파일에 대한 설명

(2)와 (3)에서 헤더 파일 안에 코드 내용이 같기 때문에 같은 헤더 파일을 사용

(1) 클래스의 선언부와 구현부를 분리하고, 모든 코드를 한 파일에 작성

CaclculatorFull.cpp

(2) 클래스의 선언부와 구현부를 헤더 파일과 cpp 파일로 나누어 프로그램 작성

Calculator.h

Calculator.cpp

(3) 클래스를 헤더 파일, cpp 파일, main 파일로 나누어 프로그램 작성

Calculator.h

CalculatorMain.cpp

main.cpp

각각의 장단점

(1) 클래스의 선언부와 구현부를 분리하고, 모든 코드를 한 파일에 작성

한 파일만 만들어도 돼서 번거로운 부분은 없었지만 간단한 문제 였음에도 가독성에는 약간의 불편함이 있었다.

(2) 클래스의 선언부와 구현부를 헤더 파일과 cpp 파일로 나누어 프로그램 작성

앞서 한 파일에 작성했던 코드의 선언부는 헤더파일이 되고, 구현부 부분은 cpp 파일이 되었다. 헤더 파일과 구현 파일로 나누어져있어 코드의 가독성과 유지 보수성을 향상시킬 수 있다. 하지만 메인 함수와 클래스 구현 부분이 같이 있어 클래스 코드와 실행 코드가 얹힐 가능성이 있기 때문에 코드 재사용성에서 약간 불리하다.

(3) 클래스를 헤더 파일, cpp 파일, main 파일로 나누어 프로그램 작성

앞에 cpp 코드를 메서드 구현 부분은 cpp 파일이 되고, 메인함수 부분은 main.cpp 파일이 되었다. 헤더 파일, cpp 파일, main 파일로 나누어져있어 이 방법 또한 코드의 가독성과 유지 보수성을 향상시킬 수 있다. 또한 코드를 재사용할 수 있고, 각 클래스를 독립적으로 관리할 수 있으며 디버깅에 용이해진다는 장점이 있다.

추가로 cpp 파일도 더 세세하게 나누게 되면 (3)에 비해서 재사용이나 디버깅 용이성은 높아지겠지만 그리 길지 않은 코드를 다 나눠 파일 수가 너무 많아져 이런 작은 문제를 풀때는 너무 많은 파일을 만들어야 해서 불편함이 있었다.

문제 해결 방법 (아이디어)

1. 클래스

- 연산을 수행하는 클래스 4개 : Add, Sub, Mul, Div
각각의 클래스는 두 개의 정수를 저장하는 변수와 연산을 수행하는 메서드 가
짐.
- setValue(int x, int y) : 입력받은 두 정수를 클래스의 멤버 변수에 저장
- calculate() : 저장된 두 정수에 대해 해당 연산을 수행하고 결과를 반환

2. 키보드 입력

- 무한 루프를 사용하여 키보드로부터 두 개의 정수와 연산자를 입력받음
- 연산자가 '+'일 경우 Add 클래스의 객체를 생성하고,
setValue() 함수를 호출해 값을 설정한 후 calculate()를 호출하여 결과 출력
- 연산자가 '-'일 경우 Sub 클래스의 객체를 생성하고,
setValue() 함수를 호출해 값을 설정한 후 calculate()를 호출하여 결과 출력
- 연산자가 '*'일 경우 Mul 클래스의 객체를 생성하고,
setValue() 함수를 호출해 값을 설정한 후 calculate()를 호출하여 결과 출력
- 연산자가 '/'일 경우 Div 클래스의 객체를 생성하고,
setValue() 함수를 호출해 값을 설정한 후 calculate()를 호출하여 결과 출력
- 4개의 연산자 이외에 문자가 입력되면 '다시 입력해주세요.'라는 메시지를 출력
하고, 다시 입력을 요청함

3. 예외

- 나누기 연산의 경우, 분모가 0일 때는 나누기 연산을 할 수 없기 때문에 '분모
에 0이 올 수 없습니다.'라는 메시지를 출력하고 다시 입력을 요청함

아이디어 평가

연산자의 경우는 문자형으로 선언했기 때문에 4개의 연산자(+, -, *, /) 이외의 문자를 입력받았을 경우 다시 입력해달라는 메시지를 출력하고 다시 입력을 받을 수 있는데 x, y에 경우는 정수형으로 선언을 했기 때문에 다른 문자를 입력하게 되면 프로그램 자체를 무한 루프로 돌게 만들어져있어 ‘두 정수와 연산자를 입력하세요>> 다시 입력해주세요’ 라는 문구가 연속으로 나오게 된다.

<‘Ctrl+C’를 통해 콘솔에서 실행 중인 프로그램을 강제 종료 하였다.>

[illegible]

Microsoft Visual Studio 디버그 콘솔

두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.
두 정수와 연산자를 입력하세요>> 다시 입력해주세요.

D:\VisualStudio\Wp#\Cplusplus\HW01\#x64\Debug\CalculatorMain.exe (프로세스 16040)이(가) -1073741510 코드(0xc000013a)와 함께 종료되었습니다.
이 창을 닫으려면 아무 키나 누르세요...

