

2023 Spring OOP Assignment Report

과제 번호 : 1-2

학번 : 20220100

이름 : 박기현

Povis ID : kihyun

명예서약 (Honor Code)

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

1. 프로그램 개요

- '+', '-', '*', '/', 'sqrt', 'square', 총 6가지 연산이 가능한 계산기 프로그램이다.
- 연산자는 char(array)형 자료형으로, 두 개의 숫자는 float형 자료형으로 입력받고, 연산자와 각 숫자 사이에는 공백 한 칸이 있다.
- '+', '-', '*', '/' 연산은 두 개의 숫자를 입력받고, sqrt와 square 연산은 한 개의 숫자만 입력받는다.
- 연산자와 숫자를 모두 입력받으면, 각 연산자에 해당하는 연산을 진행하는 사용자정의 함수를 통해 원하는 값을 출력한다.
- 계산 결과는 소수점 셋째 자리까지 반올림하여 출력하며, 정수인 경우에는 소수로 표현하지 않는다.

2. 프로그램의 구조 및 알고리즘

- 헤더 파일은 <iostream>과 <cmath>, <cstring>을 사용한다.
- char형 배열 calc_operator 변수는 연산자 변수를 나타낸다. new를 이용하여 길이가 7인 배열을 동적 할당한다.
- float형 num1, num2 변수는 연산을 진행할 두 숫자를, float형 value 변수는 연산을 진행한 최종 값을 나타낸다.
- 입력받은 연산자를 비교하기 위해 6개의 연산자 배열을 선언하고, 각각 연산자에 맞는 문자열로 초기화한다.
- cin을 이용해 calc_operator 변수에 연산자를 입력받는다. 이때 연산자가 '+', '-', '*', '/'

이면, 즉 덧셈, 뺄셈, 곱셈, 나눗셈 연산이면 두 개의 숫자를 입력받고, 'sqrt', 'square', 즉 루트, 제곱 연산이면 한 개의 숫자를 입력받는다. 연산자와 숫자는 공백 한 칸을 이용하여 구분한다.

- 연산자와 숫자를 입력받으면 입력받은 연산자에 해당하는 사용자정의 함수에 숫자를 넘겨 연산을 진행한다.
- <cstring> 라이브러리의 문자열을 비교하는 strcmp 함수를 이용하여 calc_operator 문자열과 각 연산자 배열의 문자열을 비교하고, 같으면 0이 반환되는 성질을 통해 조건문에 따라 각 연산자의 사용자정의 함수로 넘겨준다.
- 연산을 진행한 최종 값은 float형 변수이기 때문에 사용자정의 함수에 float형으로 형 변환하여 입력받은 숫자를 넘겨준다.
- 사용자정의 함수 내에서는 연산을 진행한 값을 반환한다. 이때 sqrt의 경우 <cmath> 라이브러리에 포함된 sqrt 함수를 이용한다.
- 사용자정의 함수로부터 반환된 값이 정수가 아니면 소수점 셋째 자리까지 반올림하여 출력한다.
- 출력 후, 프로그램을 종료한다.

3. 토론 및 개선

- 기본적인 cin, cout에 대해서 익힐 수 있었으며, char형 배열과 동적 할당하는 함수 new와 delete의 사용법에 대해 익힐 수 있었다. <cmath> 라이브러리에 포함된 함수 sqrt에 대해서 배울 수 있었고, 수업시간에 배운 소수점 자릿수 고정과 몇째 자리까지 고정할 것인지에 대해 구현하는 방법을 적용해볼 수 있었다.
- 처음에 덧셈, 뺄셈, 곱셈, 나눗셈 연산인 경우와 sqrt, square 연산인 경우에 따라 입력받는 숫자 개수가 다르게 구현하는 방법에 대해서 고민하는 과정이 가장 어려웠다. 이때 조건문을 활용한다면 앞에 입력받은 연산자에 따라 오류 없이 숫자를 입력받을 수 있다는 것을 알게 되었고, 실제로 해결할 수 있었다.
- 입력받은 연산자에 따라 각 연산자에 맞는 사용자정의 함수로 넘겨주기 위해서는 조건문을 활용해야 한다고 생각하였으나, 그 조건문의 조건을 어떻게 설정해야 할지 감이 잡히지 않았었다. 처음에는 연산자의 문자열은 '+', '-', '*', '/', 'sqrt', 'square'로 변하지 않는다는 것을 이용해 '+', '-', '*', '/'는 calc_operator[0]의 문자만, 'sqrt', 'square'는 서로 구분되는 calc_operator[2]의 문자만을 비교하면 해결할 수 있을 것이라 생각하였다. 하지만 코드를 실행하는 과정에서 오류는 없지만, 타인이 코드를 봤을 때 주석

이 없다면 조건문의 조건을 잘 이해하지 못할 수도 있을 것이라 생각하였다. 따라서 더 정확하게 구현하기 위해서는 `calc_operator`의 문자열을 모두 비교하는 방법으로 조건을 변경하면 좋을 것이라 생각하였다. 이때 프로그래밍과 문제해결 수업시간에 배운 문자열 비교 함수 `strcmp`가 생각났고, 이를 통해 구현할 수 있었다.

- 더 개선할 수 있는 방법으로 `char`형 배열이 아닌 `string`을 활용한다면 더 직관적으로 구현할 수 있을 것이라 생각한다.

4. 참고 문헌

- 조성현 교수님의 <객체지향프로그래밍> 수업 자료
- <https://cplusplus.com/reference/cstring/strcmp/>