**2022 Spring OOP Assignment Report**

과제 번호 : prob1

학번 : 20210273

이름 : 하태혁

Povis ID : hth021002

**명예서약 (Honor Code)**

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

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

프로그램을 하다 보면 결정해야 할 세부 사항이 많은데, 이러한 세부 사항을 처리한 방법과 이유를 보고서에 쓰십시오.

독창적인 아이디어와 추가 기능은 보너스 점수를 받을 수 있으므로, 보고서에 명확히 기재하십시오.

문제가 여러 개인 경우, 각 문제별로 정리해서 작성합니다.

각 문항별 설명은 편의를 위한 것으로, 삭제하고 제출한다.

1. **프로그램 개요**

이 프로그램은 input.txt 파일에서부터 연산자와 실수를 받아온 뒤, 입력된 연간자로 계산한 결과를 output.txt 에 출력하는 계산기 프로그램이다. 데이터 파일을 불러올 때는 파일 포인터와 <cstdio> 헤더파일에 포함되어 있는 fopen()을 사용했고 연산하는 과정에서는, float타입의 calculator() 라는 사용자 함수를 정의하여 계산한 값을 float형으로 반환하고자 하였다.

|  |
| --- |
|  |
|  |

프로그램 디렉토리 안에는 제출 코드와 test용 input.txt 파일이 존재한다. 그리고 프로그램을 실행시키면 같은 디렉토리 내부에 output.txt 파일이 생성되고 코드를 통해 나온 결과가 출력된다.

1. **프로그램의 구조 및 알고리즘**

텍스트이(가) 표시된 사진

자동 생성된 설명

파일에서 읽은 실수를 저장할 float형 변수 두 개와 연산자를 저장할 char형 변수와 계산값을 저장할 float형 변수를 선언한다.

텍스트이(가) 표시된 사진

자동 생성된 설명

fopen 함수가 input file을 잘 불러왔는지 확인하기 위해 위와 같은 코드를 작성하였다. Input.txt 파일이 존재하시 않을 시 에러 메세지와 함께 프로그램이 종료된다.

텍스트이(가) 표시된 사진

자동 생성된 설명

파일에서 한 줄 단위로 데이터를 받아오기 위해 fscanf 함수를 사용했고 각각 연산자와 실수 두 개를 받아온다. 받아온 연산자와 변수가 calculation() 함수에 입력되고 조건문 if를 통해 어떤 연산을 처리해야 하는지를 판단하고 해당 연산을 진행하여 값을 반환한다.

텍스트, 화면, 스크린샷이(가) 표시된 사진

자동 생성된 설명

계산된 값은 소수점 아래 3자리로 표현하며 계산 값이 이보다 길 경우 반올림하여 output.txt 파일에 출력한다. 반올림을 계산하기 위해 %.3f 을 사용했고 fprintf 함수를 사용했다.

1. **토론 및 개선**

C++ 코딩을 하기 전에 필요한 작업환경 구축과 C언어에 대한 전반적인 복습을 해볼 수 있었다. 또 C++에서는 헤더파일에서 <stdio.h> 대신 <cstdio>, <iostream> 등을 사용한다는 것을 배우게 되었다.

객체지향 프로그램으로 구현할 시, 계산기 기능을 캡슐화하여 하여 객체로 구현할 수 있을 것이다. 그러면 프로그램을 구성할 때 계산 알고리즘을 다시 설계할 필요없이 메소드를 불러와 해결 할 수 있을 것이기 때문에 더 간소한 코드를 구성해 볼 수 있을 것 같다.

이후 계산기 기능을 다른 프로그램에서 사용할 때도 calculaitor 함수를 정의할 필요없이 객체에서 메소드를 바로 불러올 수 있기 때문에 제작한 코드를 여러 프로그램에서 재사용할 수 있을 것이다.

1. **참고 문헌**

[https://ko.wikipedia.org/wiki/절차적\_프로그래밍](https://ko.wikipedia.org/wiki/%EC%A0%88%EC%B0%A8%EC%A0%81_%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D)

[https://ko.wikipedia.org/wiki/객체\_지향\_프로그래밍](https://ko.wikipedia.org/wiki/%EA%B0%9D%EC%B2%B4_%EC%A7%80%ED%96%A5_%ED%94%84%EB%A1%9C%EA%B7%B8%EB%9E%98%EB%B0%8D)

절차지향과 객체지향 프로그램의 차이를 명확히 인지하기 위해 두 프로그램에 대해 공부했습니다.