

# 구성원 소개

01

이강민

02

조희강

03

민준수

04

김민철

# 발표 개요



**구현 목표**



**헤더 파일과  
소스 파일의  
관계 설명**



**알고리즘  
설명**



**강조 포인트**



# | 구현 목표 설정

무제한의 입력 문자열

계산 해야 하는 입력 문자열의 길이는 제한이 없어야 한다.

덧셈, 뺄셈, 곱셈, 나눗셈 구현

입력된 식을 연산하기 위해서  
정확한 사칙연산 기능을 구현한다.



파일로부터 입력

식이 작성되어 있는 파일을  
입력 받게 된다.



스택을 이용한 후위연산 표기

입력 받은 중위표기식을  
후위표기식으로 변환해 연산한다.



# 헤더 파일과 소스 파일 관계 설명

Main.c

fileReader.c

calculator.c

calculate.h

Plus.c

Minus.c

Multiply.c

Divide.c

calculator.h

returnValue.c

파일로 받은 입력값을  
문자열로 저장

Calculator.h의 구조를  
사용하는 각 연산 함수

연산 함수를 포함하여  
계산을 수행할 수 있는  
Calculate.h 생성

수와 연산자를 구분하여  
실제 계산을 수행하는  
Calculator.c

결과 값을 출력

# 알고리즘 설명

연산 총괄



파일을 입력받아  
동적할당을 이용해  
문자열로 저장한다.

Step1

입력 값이  
올바른 표현이라면  
연산을 진행한다.

Step3

문자열의 문자들을 더블  
연결 리스트, 스택을 통해  
연산을 위한 구조로  
저장해준다.

Step2

Step4

계산된 결과 값을 출력한다.



# 알고리즘 설명

## 연산 상세

Calculator.c 에서  
스택을 이용하여  
중위표기식을  
하나씩 꺼내온다.

**Step1**

**Step2**

사칙연산의 우선순위를  
고려해 순서를 조정하고,  
괄호를 고려해  
후위표기식으로 변환한다.

후위표기식을 순서대로  
읽어가며 연산함수를  
이용하여 연산을 진행한다.

**Step3**

**Step4**

만약 0과 9사이의 수와  
연산자가 아닌 문자가  
들어오거나 연산자가  
연속해서 들어오면 에러  
메세지를 출력한다.



# 알고리즘 설명

Plus함수

First수와 second수의  
소수점 뒤자리의 자릿수를  
비교하여 자릿수가 큰 수와  
작은 수로 구분한다.

Step1

현재 자릿수의 연산 결과가  
100이상일 경우  
앞 자릿수에 1을 추가로  
계산한다.

Step2

가장 작은 자릿수의 수부터  
상대의 같은 자릿수의 수와  
덧셈을 한다.

Step3

Step4

이후 소수점 앞자리끼리의  
연산도 같은 과정으로  
이루어져 결과값으로  
반환한다.



# 알고리즘 설명

Minus함수

First수와 second수의  
소수점 뒤자리의 자릿수를  
비교하여 자릿수가 큰 수와  
작은 수로 구분한다.

앞 수의 절댓값이  
더 작은 경우 앞 자리 수에서  
1을 감소시킨다.

Step1

Step2

Step3

Step4

자릿수가 작은 수에 '0'을  
추가해 자릿수가 큰 수와  
자릿수를 동일하게 맞춰주고  
같은 자릿수끼리 뺄셈을 한다.

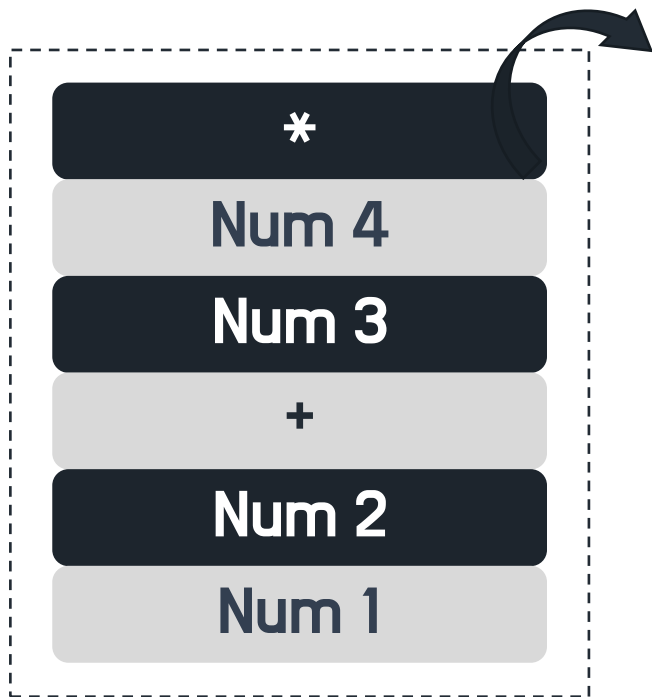
이후 소수점 앞자리끼리의  
연산도 같은 과정으로  
이루어져 결과값으로  
반환한다.



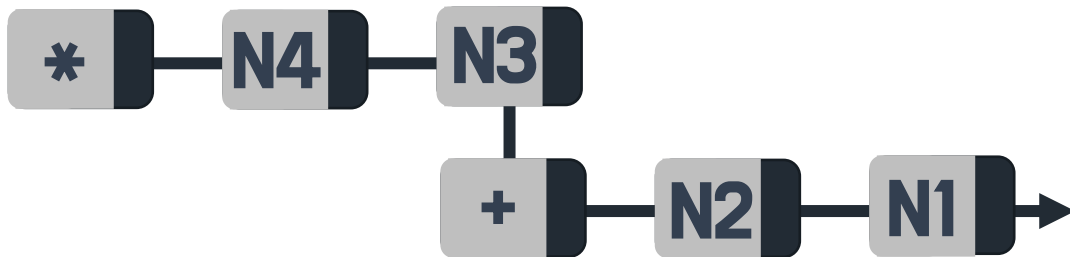


# 알고리즘 설명

Stack, Double linked list



Stack



Double linked list

ex) 123.45

First



Second



# 강조 포인트

## 추가적인 기능



### 현실 속 계산기와의 유사성 확보를 위한 기능 추가

계산기를 구현한다고 하나, 현실에서 사용하는 계산기와는 차이가 존재한다. 그래서 추가적인 기능을 제공하여 실제 계산기와 유사한 형태의 계산기 코드를 구현하고자 하였다.



### 잠셈과 나눗셈 연산 기능



### Digit copy 기능

