

두 스택을 이용한 중위표현식 계산 함수

괄호로 묶인 중위표현식을 2개의 스택을 (하나는 피연산자용, 하나는 연산자용) 이용하여 결과값을 계산하고, 그 결과를 돌려주는 `infix2StackEval()` 함수를 다음과 같이 작성한다. 각 토큰(문자)가 처리될 때 마다 두 개의 스택을 출력한다.

1. 교재에 있는 스택 함수의 자료구조를 참고. 중위표현식의 최대 길이는 100으로 제한함.
2. 연산은 모두 괄호로 묶임: 예 : $(((2 + 3) / 5) * (9 - 6))$. 콘솔창에서 사용자로부터 입력 받을 수 있도록 함.
3. 피연산자용 스택에는 숫자만 들어가고, 연산자용 스택은 연산자만 들어감. 즉 괄호는 스택에 들어가지 않음.
4. 최종 결과물 출력 시 숫자 하나만 출력.
5. 헤더 파일은, `#include <stdio.h>`, `#include <stdlib.h>`, `#include <math.h>` 세 개 만 이용할 것.
6. 숫자의 입력은 한 자릿수 이상을 처리 해야함. 즉, 10, 100, ...을 처리할 수 있어야 함.
7. 그 외 수식에 문제가 있는 경우는 고려하지 않음.(예 : 괄호 개수 부족, 0으로 나누기 등)

실행 예

1. ./infix2StackEval

중위표현식 (((2 + 3) / 5) * (9 - 6))

피연산자 스택:

2 2 2 5 5 5 1 1 1 1 1 1 3

[illegible]

연산자 스택:

[illegible]

최종결과 : 3

2. `./infix2StackEval`

중위표현식 $((2 + 3) / (5 - 4)) * (9 - 6))$

피연산자 스택:

연산자 스택 :

연산자 스택 :

[illegible]

최종결과 : 15

3. `./infix2StackEval`

중위표현식 $((2 + 3) / (5 - 4)) * (9 - 6)$

피연산자 스택:

[illegible]

연산자 스택 :

$$\begin{array}{ccccccccc} & & & & & & - & - & \\ & + & + & & / & / & / & / & / \\ == & == & == & == & == & == & == & == & == \end{array}$$

최종결과 : 7

프로그램 작성 환경 조건

1. 학과 리눅스 서버 vi에디터를 이용하여 작성한 후 turnin에 제출한다.
2. gcc 컴파일러를 이용하여 컴파일한다.

주의 사항

1. 미 제출시 0 점 처리, 부정행위 적발 -10점처리함(추후 제출하는 숙제에서 감점 처리함)