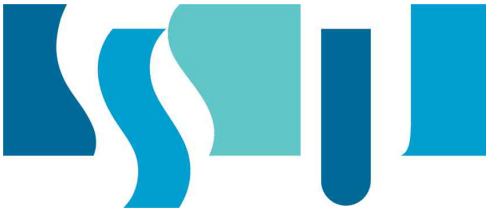


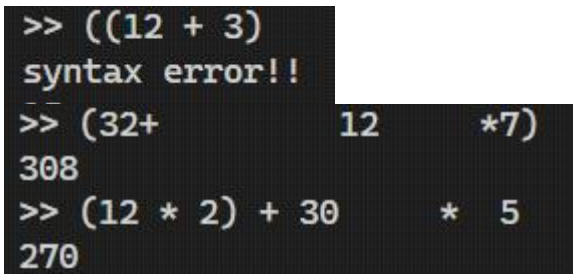
# 프로그래밍언어

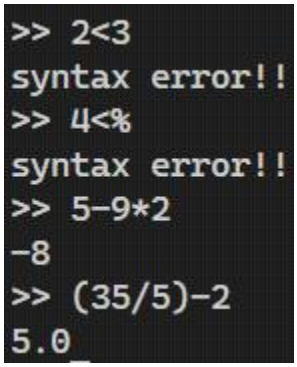
## [과제 2]

## RD Parser

| | |
|---|---|
| **과목명** | 프로그래밍언어 |
| **교수명** | 유진 |
| **분 반** | 나반 |
| **학 과** | 컴퓨터학부 |
| **학 번** | 20211799 |
| **이 름** | 유진 |
| **제출일** | 2023.05.03 |

실행 결과 :




비교연산에 대한 처리 (<bexpr> → <aexpr> <relop> <aexpr>)를 구현하지 못하였습니다.

함수 및 구현 방법에 대한 설명은 cpp를 기준으로 설명합니다.
입력된 수식을 토큰으로 분리한 후, 토큰에 따라 수식 계산을 수행하여 결과를 출력합니다.

CPP에서는 큐(queue)를 사용하여 토큰을 저장하고, 수식 분리에 사용되는 dec() 함수와 tokenParse() 함수, 계산에 사용되는 aexpr(), term(), factor(), number() 함수 등이 정의되어 있습니다. python에서는 deque, java에서도 큐를 사용하였습니다.

수식 계산은 수식의 우선순위에 따라 괄호 안의 수식부터 계산하며, 괄호가 여러 겹일 경우 내부부터 차례대로 계산합니다. 각 연산자에 따라 덧셈과 뺄셈이 우선적으로 계산되며, 곱셈과 나눗셈은 순서대로 계산됩니다.

또한, 에러 처리도 구현되어 있습니다. 입력된 수식에 문제가 있거나 처리되지 않은 토큰이 남아 있을 경우, "syntax error!!"를 출력하고 프로그램을 종료합니다.

이 코드에서는 getline() 함수를 이용하여 사용자로부터 수식을 입력받습니다. 수식을 입력하지 않으면 프로그램이 종료됩니다.

코드의 실행 흐름은 다음과 같습니다.

1. 사용자로부터 수식을 입력받는다.
2. 입력된 수식을 토큰으로 분리한다.
3. 분리된 토큰을 이용하여 수식을 계산한다.
4. 결과를 출력한다.

5. 프로그램을 종료한다.

다음은 함수에 대한 설명입니다.

error(): 문법 오류가 발생한 경우 호출되는 함수로, "syntax error!!"를 출력하고 큐를 비웁니다.

dec(string str, int index): <dec> → 0 ¦ 1 ¦ 2 ¦ 3 ¦ 4 ¦ 5 ¦ 6 ¦ 7 ¦ 8 ¦ 9을 구현하였으며, 문자열에서 현재 인덱스부터 시작해서 숫자를 읽어 정수로 변환하여 큐에 저장하고, 읽은 숫자의 길이를 반환합니다.

tokenParse(string str): 입력받은 수식을 토큰으로 분리하여 큐에 저장합니다. 숫자인 경우 dec() 함수를 호출하여 처리하고, 연산자 및 괄호인 경우 바로 큐에 저장합니다. 그 외의 경우에는 문법 오류가 발생한 것으로 처리합니다.

expr(): 수식을 처리하는 함수입니다. 이 함수는 다른 수식 함수를 호출하며, 수식에 따라 결과값을 반환합니다

bexpr(): 수식의 비교 연산자를 처리하는 함수입니다. 이 함수는 다른 수식 함수를 호출하며, 비교 연산자에 따라 결과값을 반환합니다.

relop(): 다음 토큰이 비교 연산자인지 확인하는 함수입니다.

aexpr(): <aexpr> → <term> {* <term> ¦ / <term>}를 구현하였습니다. 수식에서 곱셈과 나눗셈을 처리합니다. term() 함수를 호출하여 반환된 값을 이용해 곱셈과 나눗셈을 처리하고, 그 결과를 반환합니다.

term(): <term> → <factor> {+ <factor> ¦ - <factor>}를 구현하였습니다. 수식에서 덧셈과 뺄셈을 처리합니다. factor() 함수를 호출하여 반환된 값을 이용해 덧셈과 뺄셈을 처리하고, 그 결과를 반환합니다.

factor(): <factor> → <number> ¦ (<aexpr>)를 구현하였습니다. 수식에서 가장 우선순위가 낮은 항목을 계산합니다. 음수인 경우와 숫자인 경우 처리가 달라지며, 괄호가 있는 경우에

는 aexpr() 함수를 호출하여 처리합니다.

number(): <number> → <dec> {<dec>}를 구현하였으며, 큐에서 정수로 된 숫자를 읽어 반환합니다.

lex(): 큐에서 토큰을 하나 읽어 반환합니다.

nextValue(): 큐에서 다음 토큰을 반환합니다.

queueClear(): 큐를 비웁니다.

main(): 사용자로부터 수식을 입력받고, tokenParse() 함수를 호출하여 큐에 토큰을 저장합니다. 그 다음 aexpr() 함수를 호출하여 수식을 계산하고, 그 결과를 출력합니다. 입력된 수식이 없는 경우 프로그램을 종료합니다.