

과제2

Make_postfix(istream& ins) 함수를 먼저 구성 해주었다.

벡터로 선언된 이 함수는 중위 표기법을 후위 표기법으로 변경해 벡터를 만들어 주는 것이다. 그리고 함수 안 변수로 참조자를 통해 ins라는 변수를 stream 형식으로 받아오는데 나는 메인 함수에서 string형식으로 받아와서 라이브러리 sstream을 넣어서 istrstream을 활용해 스트림형식으로 변환해주는 과정을 거쳤다.

이 함수 안에 postfix 결과를 저장할 새로운 벡터 result_postfix를 만들어 한글자씩 입력을 받을 임시 변수 ch_temp 변수를 통해 처리를 해주었다. 처리 해주는 것은 자료들을 보고 괄호 같은 경우는 Lparen, Rparen 를 구분해주고 마지막에 갯수도 비교해 괄호가 알맞게 열고 닫혔는지 까지 확인하는 것을 구현했다.

그리고 숫자를 구분하기 위해 isdigit 함수를 사용했는데 라이브러리 ctype를 넣었다.

isdigit 함수란 char타입이 숫자로 이루어지면 true를 반환하는 함수이다.

그리고 연산자 우선순위에 따라 리턴값이 다른 int형 함수 compare_op를 사용해서 연산자 처리까지 해주게 구현했다.

evaluate_stack 함수는 스택에 있는 두 숫자를 꺼내와서 연산자 하나를 계산하는 함수이다.

예외처리를 우선 해주었고 숫자 두개 변수를 num1, num2 로 선언해주고 top, pop을 반복해 초기화해주었고 계산 결과 변수 cal를 선언해 연산자에 따라 계산되게 해주었다. Pow() 제공해주는 함수를 쓰기 위해 라이브러리 'Cmath'를 넣어주었다. 그리고 계산된 것이 다음 숫자와 계산되어야 하기에 push해주게 마무리 해주었다.

Evaluate_postfix() 함수는 위 evaluate_stack 함수를 써서 최종 계산값을 도출하는 함수이다.

Auto : 자료형에 맞게 쓸 수 있는 것 -> 템플릿과 비슷함

새로운 스택 cal_number: 최종계산값 을 선언해주고 if (cal_number.size == 1) 즉 하나만 남았을 때 계산이 끝난 것이므로 return cal_number.top() 반환하고 끝이다.

이 과제를 할 때 벡터와 스택 함수에 익숙치 않아서 어려웠는데 자료검색을 해보니 함수

명(매개변수) 매개변수 자리에 (istream& ins)가 검색해도 잘 나오지 않았고 참조자라 쓴 의도를 알고 입력 클래스 관련 자료에 뜨길래 공부해보니 입력을 다양한 방식으로 받을 수 있는 좋은 변수라는 것을 알게 되었고 메인에 string으로 입력받는 식으로 코드를 짰었는데 스트림의 string과는 다르다는 것을 알게 되어 stringstream을 사용해서 해결하는 과정을 거쳤다. 그 이후에는 후위표기 계산기 자료가 많아 원활하게 구현했다. 2번 과제를 하면서 어려움을 많이 겪었는데 괄호까지 구현해서 좋았지만 두자리 수를 구현하지 못하였다. 두자리 수를 구현하기 위해 숫자 간 띄어쓰기를 구분해서 띄어쓰기가 없으면 두자리숫자 혹은 세자리 숫자 등등으로 인식하게 하고 싶었으나 하지 못해 아쉬웠다. 그래도 입력, 참조자 등 교육 받은 것 이외에 다른 것을 많이 배우게 되어 좋았다.