

Programming Language

Assignment # 2

20150291 하현수

1. 오류 검색 방식

대부분의 오류 검색은 InfixToPostfix.cpp 파일 안에 구현되어 있으며 에러는 크게 6가지로 나뉩니다.

첫째로는 피연산자 자리에 정수가 아닌 실수가 들어갔을 때, 둘째로는 괄호가 잘못 사용되었을 때, 셋째로는 기호 사용이 잘못되었을 때, 넷째로는 정의되지 않은 연산자를 사용했을 때, 다섯번째로는 피연산자 자리에 -가 연속 입력되었을 때, 마지막으로는 숫자뒤에 문자 또는 기호가 공백 없이 들어가 있을 때입니다.

각각의 에러 메시지들은 errnum_string 배열에 1,2,3,4,5,6 인덱스 순서대로 저장되어 있으며, errnum을 바꿔줌으로써 다양한 에러를 처리합니다. 다음은 에러 처리를 위한 함수들입니다.

Is_val() : 변수를 처리하는 함수로, 만약 문장안에 문자나 숫자 이외에 것이 나온다면 기호 사용이 잘못되었음을 의미하는 3번 에러 상태로 만듭니다.

Is_con() : 상수인를 처리하는 함수로, 만약 문장안에 .이 나오면 실수로 판단하여 에러 상태 1로 바꾸고, 숫자가 아닌 것을 상수로 사용할경우 6번 에러 상태로 만듭니다.

Is_Op() : 연산자를 처리하는 함수로, 만약 연산자 뒤에 공백이 아닌 다른 문자가 올 경우 에러 상태 4번을, 만약 해당 문자가 괄호일 경우 2번 에러 상태로 만듭니다.

Is_minus_con() : -로 시작되는 상수를 처리하는 함수로, 만약 -가 문장 뒤에 올경우 에러 상태 5번을, -뒤에 문자가 나올 경우 에러 상태 3번을, 그리고 -뒤에 아무 상수도 안 적혀 있을 경우 1번 에러 상태로 만듭니다.

Check_err_after_program() : 프로그램과 \n 사이에 공백 이외에 다른 문자나 상수가 들어있을 경우 2번 에러 상태로 만듭니다.

Printerror() : 에러 상태에 따라서 errnum_string에 있는 에러 메시지를 출력해 주는 함수입니다.

그외에도 make_Postfix, Is_open_parentheses 등 함수내에서도 자체적으로 괄호 개수, 변

수 상수 검사를 통해서 에러를 핸들링 합니다.

2. 중간코드생성 분석

중간코드 생성은 우선 표기방식을 InfixToPostFix.cpp에서 후위 표기로 변경한 다음 PostfixToIntermediate.cpp에서 중간 코드를 생성합니다.

Make_Intermediate() 함수에서 중간코드를 생성하게 됩니다.

먼저 함수 초입에서, 리스트가 빌 때까지 코드를 한 줄씩 읽으면서 처리를 하게 됩니다. 먼저 Program의 맨 앞에서는 begin을 추가하고, 상수 혹은 변수일 경우 각각 is_con() , is_val() 함수에서 처리하여 Push중간 코드를 추가하고, 연산자일 경우에는 is_op() 함수에서 처리하여 연산자 이름을 중간코드에 추가합니다.

3. 인터프리터

인터프리터는 Interpreter.cpp에 정의되어 있으며, 연산 결과를 도출합니다. 중간코드를 begin을 한줄씩 읽어나가면서 begin을 만날경우 스택과 undefined와 에러를 초기화 합니다. 또한 end없이 begin을 다시 만날 경우 에러로 처리합니다.

Push를 만났을 경우에는 뒤에 오는 값이 변수인지 상수인지 구분한 뒤 스택에 push합니다. End를 만났을 경우, 변수 값이 정의되지 않았거나 프로그램 자체가 잘못됐을 경우 에러로 처리합니다. 연산자를 만날 경우, IF일 경우 cal1 > 0일 경우 cal2를 push하고 아닐 경우엔 0을 push합니다. MINUS일 경우 cal1 - cal2를 스택에 push합니다.

4. 수정 및 구현한 부분

InfixToPostFix.cpp에서 표기 방식을 변경합니다.

후위 표기 방식으로 변환하는 과정은 is_open_parentheses() 함수에서 처리합니다.

is_open_parenthese 함수는 열린 괄호를 만났을 때 작동하는 함수로, 기존 방식은 TERM, FUN, TERM으로 구성되어 있었지만, 본 과제에서 수정된 명세로 인해 FUN, TERM, TERM 방식으로 순서를 바꿔 구현하였습니다.

FUN에서는 is_op함수를 통해 연산자를 우선 처리하고, 그 후에 isalpha, isdigit, is_con is_val등을 통해서 TERM에 있는 변수나 상수 값들을 처리하는 방식으로 구성되어 있습니다.

이를 통해, 후위 표기 방식으로 변환된 식을 구할 수 있습니다.