

2023 Spring OOP Assignment Report

과제 번호 : 1-3

학번 : 20220826

이름 : 김민서

Povis ID : kimminseo

명예서약 (Honor Code)

나는 이 프로그래밍 과제를 다른 사람의 부적절한 도움 없이 완수하였습니다.

I completed this programming task without the improper help of others.

1. 프로그램 개요

- 본 프로그램은 진법 변환 및 계산기를 구현한 것으로, 사용자로부터 연산자, 피연산자를 입력받아 연산 결과를 출력한다. 이때 피연산자는 2, 8, 16진수 중 하나로 주어지며, 연산 결과는 10진수로 표시한다.

2. 프로그램의 구조 및 알고리즘

- 프로그램 설명

Main 함수는 사용자로부터 operator 및 operand 1, 2를 입력받는다. 입력받은 operator에 대응되는 함수를 호출하여 연산 결과를 얻고 출력한다. 이 때 피연산자를 decimal로 변환하여 인수로 넘긴다. 잘못된 입력의 경우(operand 및 operator 입력 잘못됨), 에러 메시지를 출력하고 nonzero return code를 반환한다.

□ 변수 설명:

op, a, b: 입력 받은 operator / operand 1 / operand 2

validOp: op가 '+', '-', '*', '/' 중 하나에 해당하면 참이다. 예외 처리에 사용된다.

a_converted, b_converted: 10진수로 변환된 a, b. (respectively)

나눗셈의 경우, 연산 결과가 실수로 표현될 수 있다. 정수로 나누어떨어지는 경우, 소수점 아래 숫자를 표기하지 않는다. 아닌 경우 소수점 아래 3번째 자리에서 반올림하여 나타낸다.

convertToDecimal 함수는 입력받은 string을 parse하여 어떤 진법으로 표현된 수인지 판별한다. 이후 10진수로 변환한다.

진법 판별 방법은, string의 첫 문자가 0임을 확인한 이후, 다음 문자가 b이면 binary, X이면 hexadecimal, 바로 digit이 나타나면 octal이 된다.

진법 변환은 2, 8, 16진수에서 공통된 알고리즘을 사용하며, 이는 다음과 같다.

□ 변수 설명:

src: 변환할 수, string type.

base: 진수 / followedByPrefix: src에서 prefix가 제거됨 (0X, 0b, 0)

accumulated: 누적하여 값을 더한다. 최종적으로 이 값이 반환된다.

digitValue: n진법 digit을 10진수 value로 변환하여 저장한다.

- 변환할 수의 높은 자리에서부터 digit을 하나씩 취한다. 해당 digit에 상응하는 10진수 숫자를 얻어(convertBaseDigitToDecimalInt 함수) accumulated 변수에 더한다. 더하기 전에, 이전 연산 결과를 shift한다. 이는 기존 연산 결과에 변환하는 진법 base를 곱하는 것으로 행해진다.

(binary: 곱하기 2)

(octal: 곱하기 8)

(hexadecimal: 곱하기 16)

convertBaseDigitToDecimalInt 함수는, digit 하나와 그 base(진법)를 인수로 받는다. 2~16진수에서 10진수로의 변환 방법은 동일하다.

- Ascii code '0', '9' 사이 => '0'과의 차이를 구한다.

ascii code 'A', 'F' 사이 => 'A'와의 차이를 구하여 10을 더한다.

예외 처리는 오류 메시지 출력 후 nonzero return code 반환하도록 구현하였다.

- 피연산자의 prefix가 0X, 0b, 0 중 하나가 아닐 경우 예외 처리
- 주어진 base의 digit을 decimal number로 변환할 때, 주어진 진법의 수가 아닌 경우 / parsing 불가능한 경우에서 예외 처리
- 피연산자의 prefix만 있는 경우 (예시: "+ 0X1F 0X") 예외 처리

3. 토론 및 개선

- n진법을 10진법으로 변환하는 방법을 익혔다.
- switch statement의 case 내에서 변수를 선언하는 경우는 반드시 중괄호쌍으로 block statement를 만들어 그 내부에서 선언해야 한다. 그렇지 않았더니 컴파일 오류가 발생하였다.
- 피연산자로 음의 값 및 실수 또한 받을 수 있도록 개선하면 좋을 듯하다.

4. 참고 문헌

