

# Week 3

- 주의 사항: 부정행위 금지, STL 사용 금지(string 사용가능), 배열 또는 링크드 리스트를 이용하여 구현할 것.
- 표준 입출력 사용을 권장 (C는 scanf / printf, C++은 cin / cout)

## 문제 2

우리가 일반적으로 수식을 표기할 때, 다음과 같이 표기한다.

$$A+B$$

$$2 + 3 * 5$$

이와 같이 (피연산자)(연산자)(피연산자)의 순서로 두 피연산자 사이에 연산자를 표기하는 방법을 중위표기법이라고 부른다.

그런데 컴퓨터에서 중위표기 수식을 순서대로 계산할 경우, 연산자의 우선순위를 고려하지 못해 애로사항이 생기게 된다. 예를 들어  $2 + 3 * 5$ 의 경우,  $*$  연산이 우선순위가 있지만  $+$ 가 앞에 있기 때문에 순서대로 계산하는 컴퓨터에서는 이를 적절하게 처리하기가 힘들다.

이 때문에 컴퓨터 프로그램에서는 수식 계산을 쉽게 하기 위하여 중위 표기된 수식을 다음과 같이 변환하여 사용한다.

$$2\ 3\ 5\ *\ +$$

이처럼 (피연산자)(피연산자)(연산자)의 순서로 연산자를 피연산자의 뒤에 표기하는 방법을 후위표기법이라고 부른다.

인하는 이런 후위 표기된 수식이 입력되었을 때, 연산을 수행하는 프로그램을 만들려고 한다.

## 입력

표준 입력으로 다음과 같이 주어진다.

첫째 줄에 주어지는 수식의 수  $N$  ( $1 \leq N \leq 10,000$ )이 주어진다. 둘째 줄부터  $N$ 개의 줄에는 각각 후위표기법으로 적힌 수식이 띄어쓰기 없이 주어진다. 이 때 수식은 정수( $1 \leq N \leq 9$ )와 연산자( $+, -, *$ )로만 이루어진다. (단 주어지는 수식의 길이  $P$  ( $3 \leq P \leq 100$ ))

## 출력

출력해야 하는 후위 연산식이 주어질 때마다 그 결과를 한 줄에 하나씩 출력한다.

### 예제 입출력 1

예제 입력	예제 출력
5	8
35+	12
34*	5
23+	11
23*5+	7
452-+	

### 예제 입출력2

예제 입력	예제 출력
4	-33
36+67*-	-24
378+-2*	27
357+2*+	12
3548--+	

### <Hint>

- 입력 받은 string s의 길이는 **s.size()** 또는 **s.length()**로 알 수 있습니다.  
string s = "hello";  
int size = s.size(); // size의 값은 5가 된다.
- 입력 받은 string s의 특정위치 문자는 **s.at(1)** 또는 **s[1]** 방법으로 접근할 수 있습니다.  
string s = "hello";  
s.at(0); // 'h'  
s.at(1); // 'e'
- char 형 숫자는 **char '0'**을 빼줌으로써 int로 바꿀 수 있습니다.  
string s = "58";  
int a = s[0] - '0'; // a는 int 5  
int b = s[1] - '0'; // b는 int 8