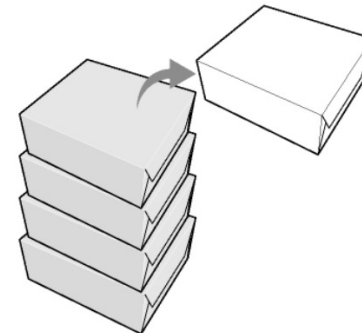
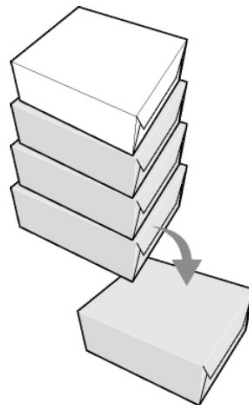


스택

스택의 개념

- 스택(Stack)
 - 일반적인 의미
 - 쌓다, 더미
 - 자료구조에서의 의미
 - 선형 자료구조 & LIFO
 - LIFO(Last-In-First-Out)
 - 후입선출(後入先出)
 - 사용 예
 - 택배에 사용되는 상자 더미(스택)

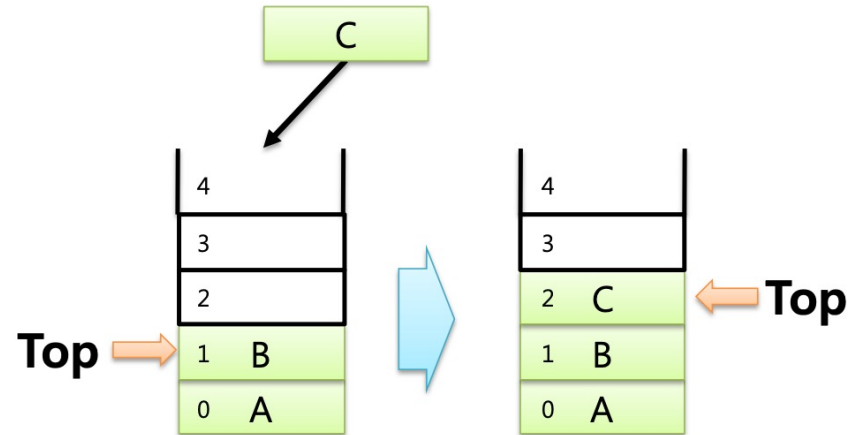


스택의 개념

- 3가지 연산

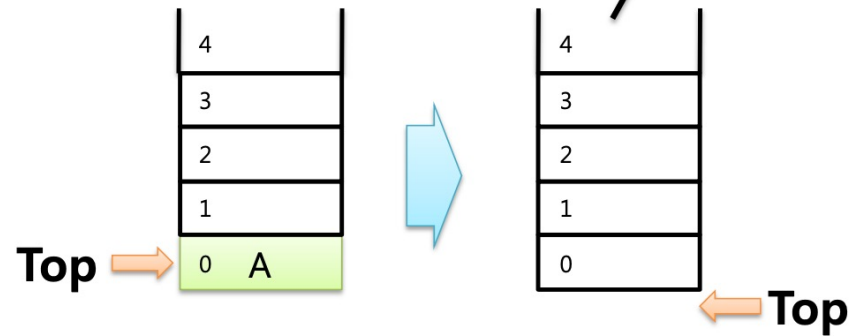
- 푸시(Push)

- 넘침(Overflow) 현상



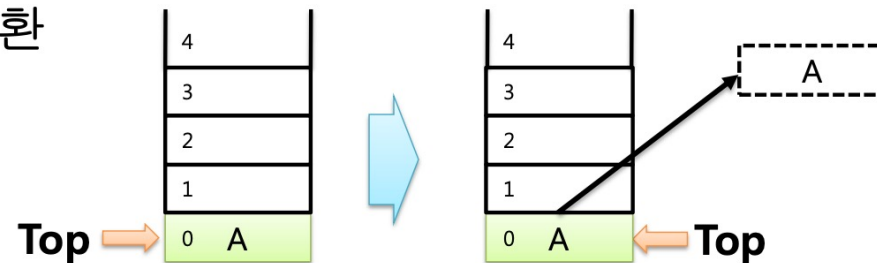
- 팝(Pop)

- 부족(Underflow) 현상

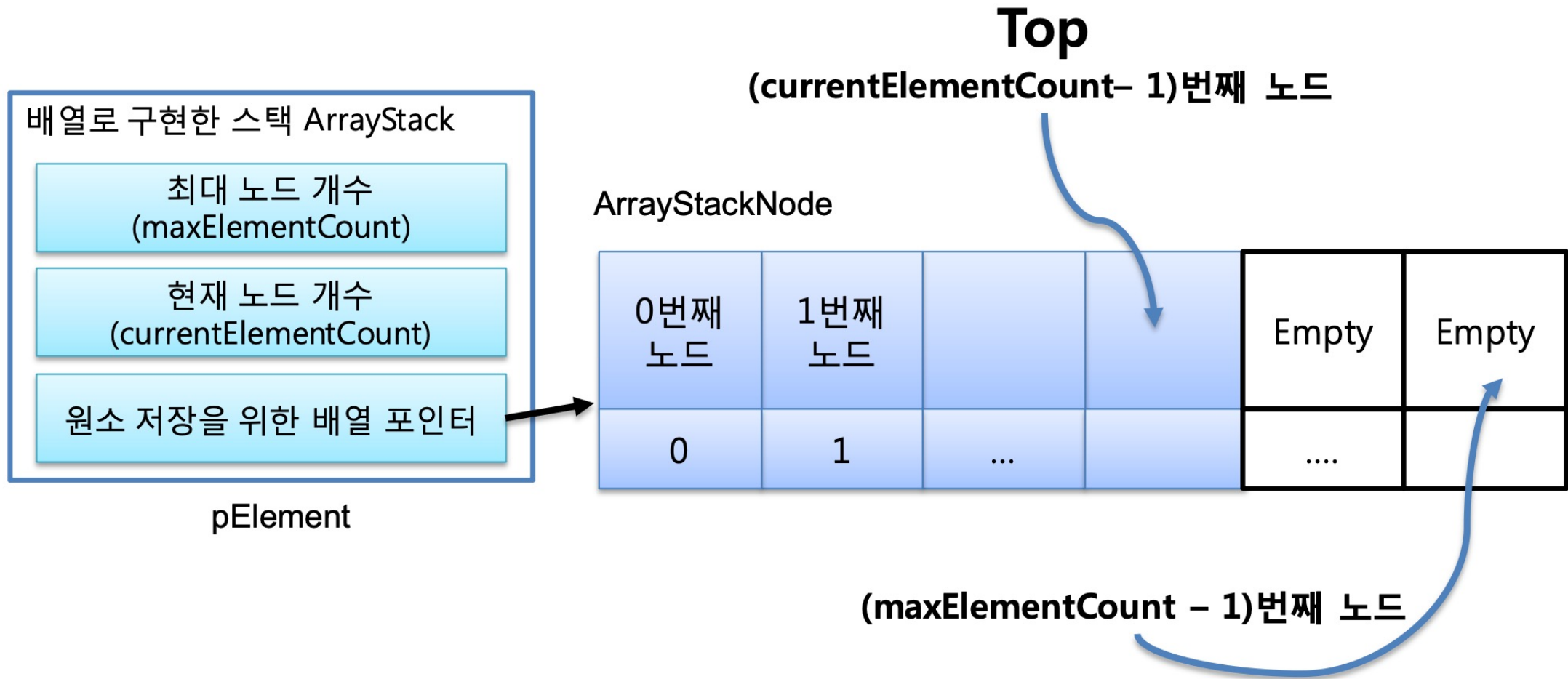


- 피크(Peek)

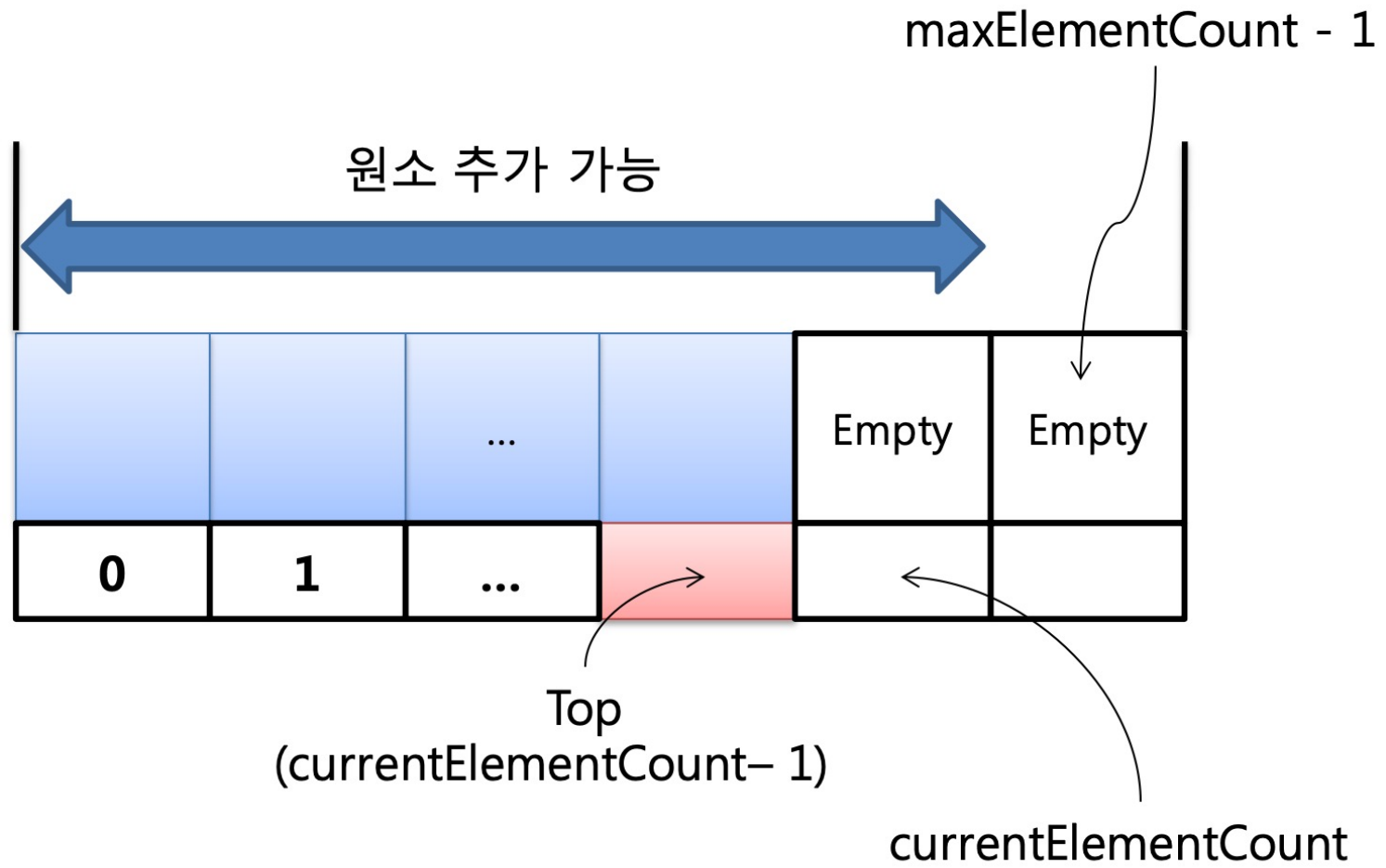
- 스택의 맨 위에 있는 원소를 반환
(스택에서 제거하지는 않음)



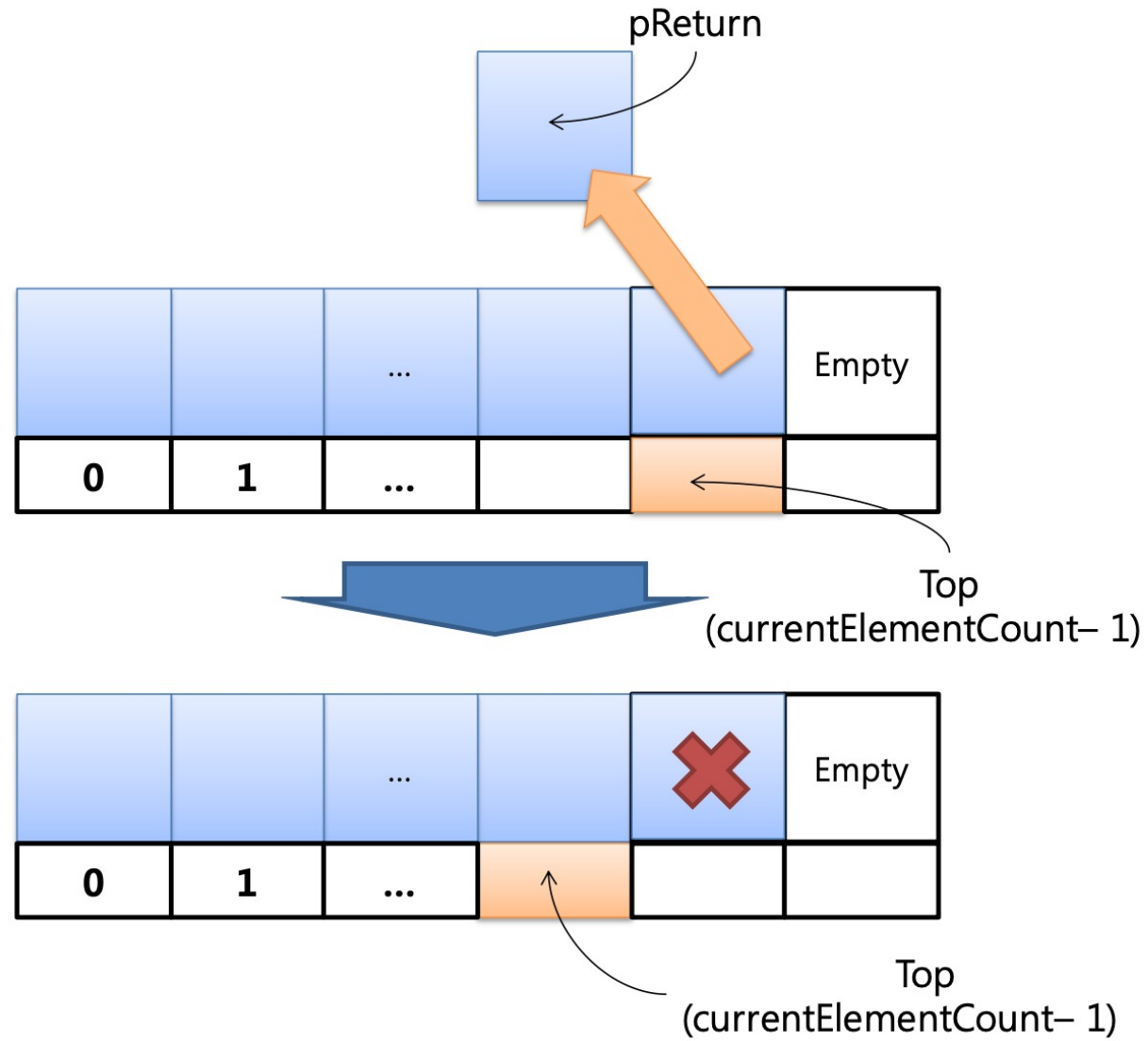
배열로 스택 구현하기



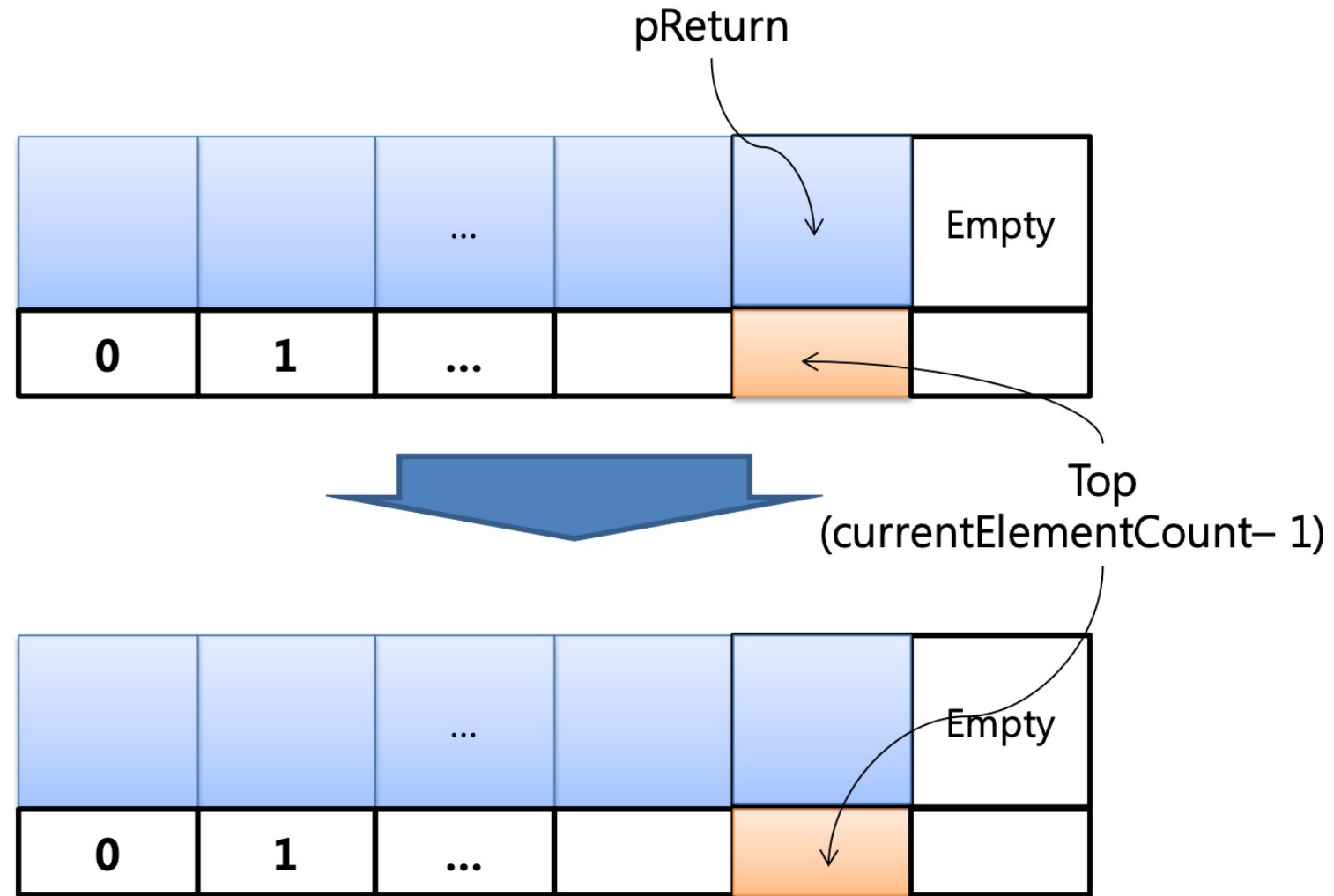
push



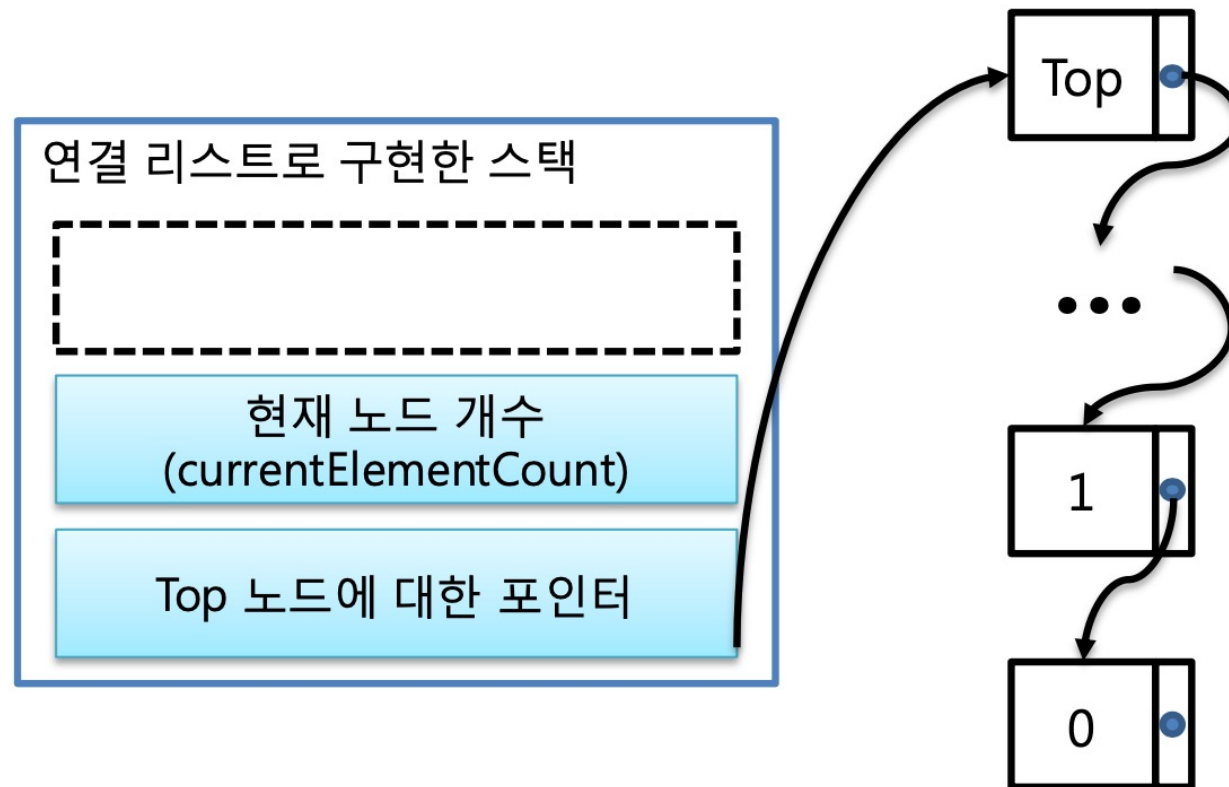
pop



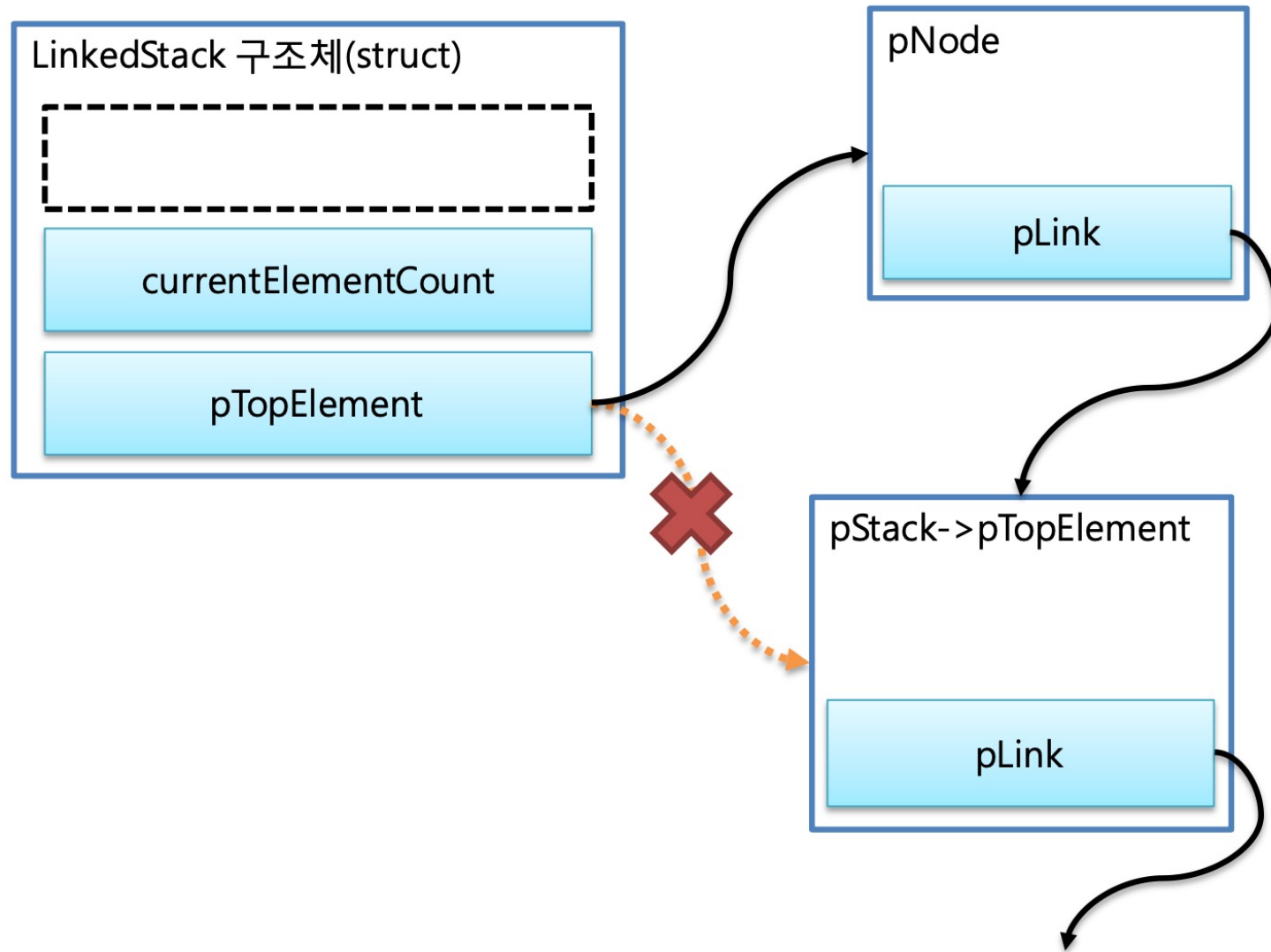
peek



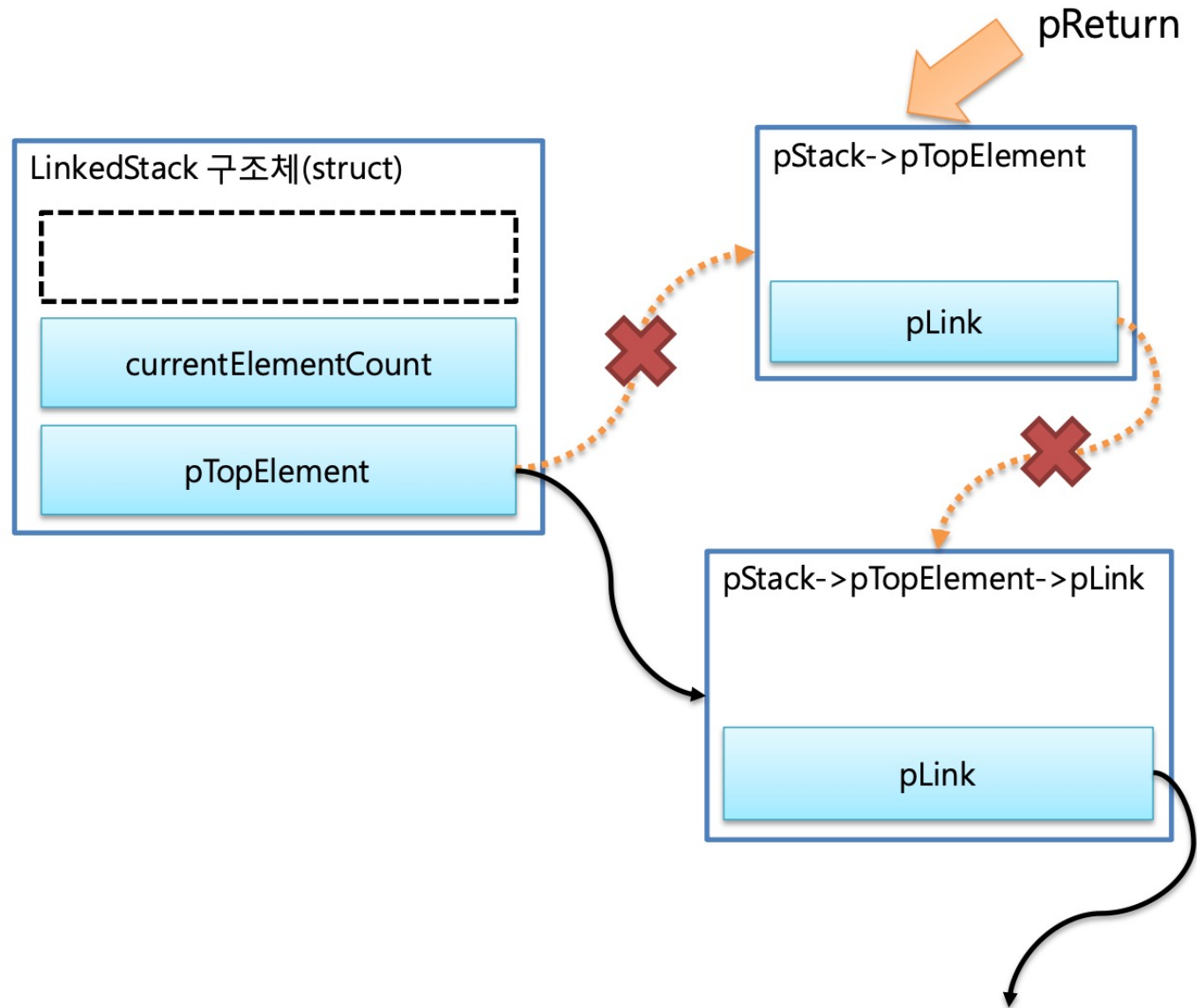
연결 리스트로 스택 구현하기



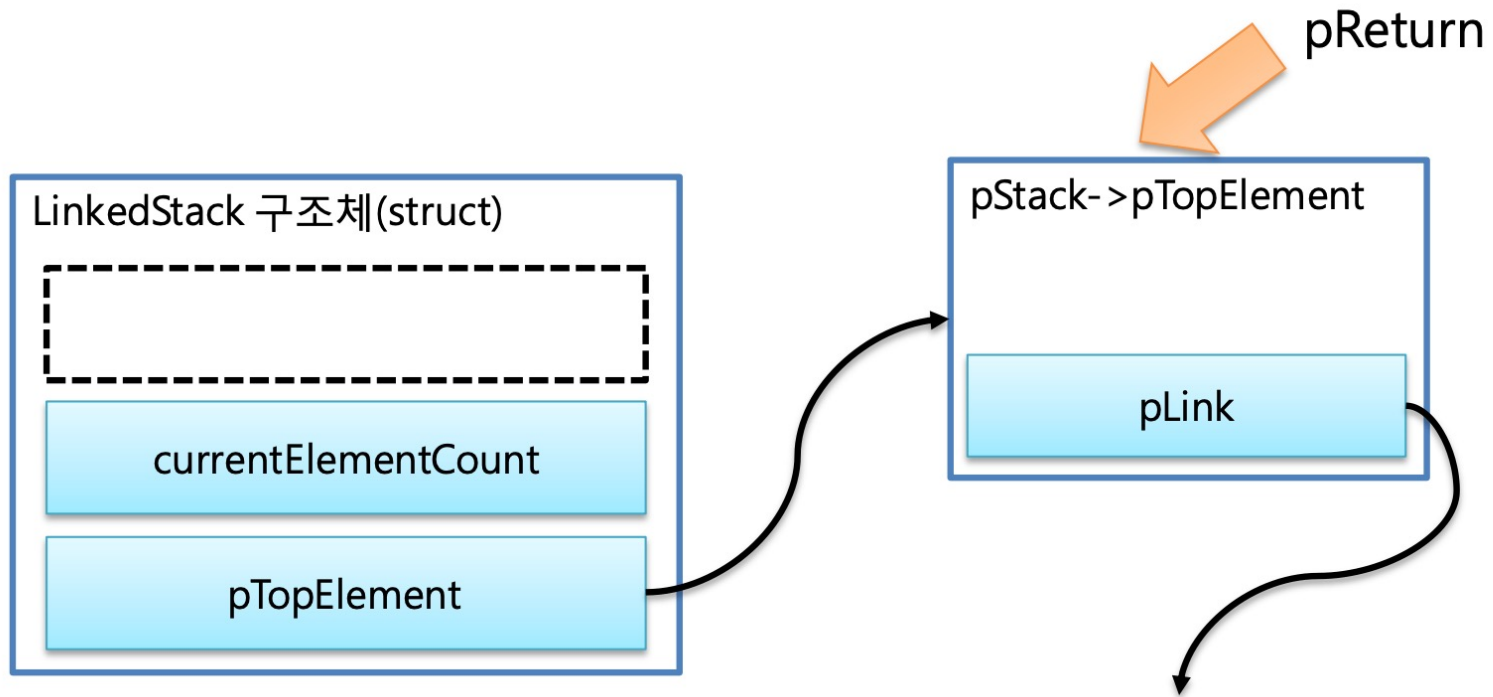
push



pop

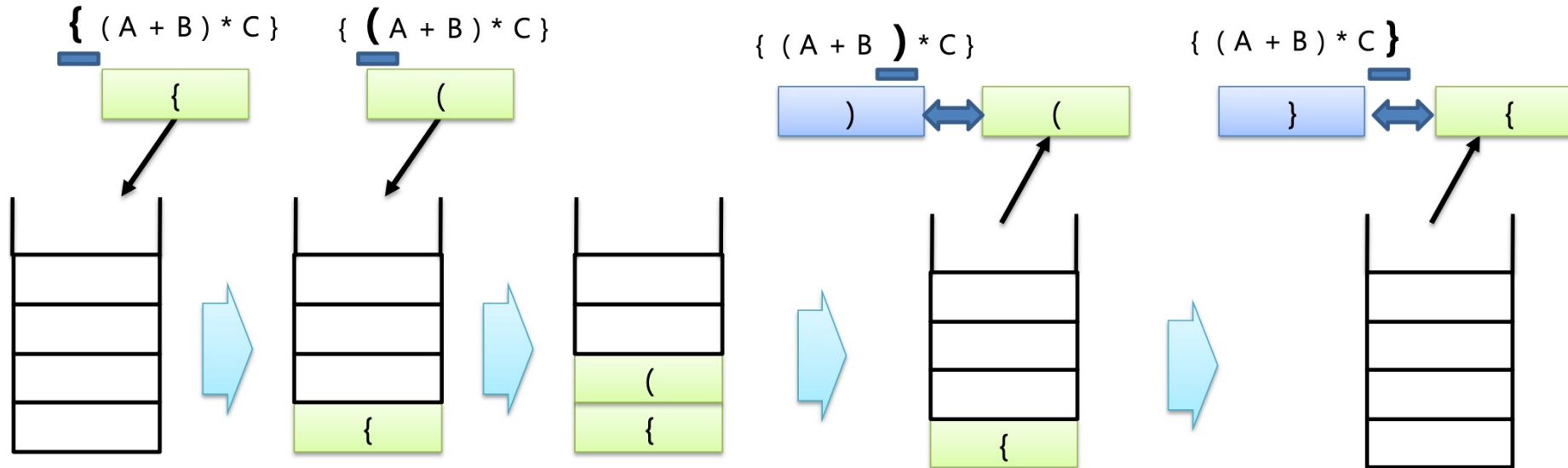


peek



수식 괄호 검사 하기

- { (A + B) * C }



$((A * B) / C) - \{ (D + E) \&\& (F - G) \}$
 $((A * B) / C - \{ (D + E) \&\& (F - G) \})$

수식 괄호 검사 하기

```
checkBracketMatching( expression) {  
    result ← 성공  
    while (expression의 끝이 아닌 경우 && result != 오류) {  
        symbol ← expression의 다음 글자  
        switch( symbol ) {  
            case '(': case '[': case '{':  
                symbol을 스택에 푸시  
                break;  
            case ')': case ']': case '}':  
                if (스택이 공백 상태) {  
                    result ← 오류  
                }  
                else {  
                    checkSymbol ← 스택에서 팝  
                    if (symbol과 checkSymbol이 쌍이 맞지 않는 경우) {  
                        result ← 오류  
                    }  
                }  
                break;  
        }  
        if (스택이 비어 있지 않다면) {  
            result ← 오류  
        }  
    }  
    return result  
}
```