

Lab#8 Tree Build and Traversal

1. Node Creation:

```
Struct node {  
    Char data;          Int prio;  
    Struct node *left; // left link    Struct node *right; // right link  
}
```

2. Precedence Table (연산자 우선순위 테이블)

char prec[5][2] = {'^' 3, '*' 2, '/' 2, '+' 1, '-' 1};

	^	*	/	+	-
	3	2	2	1	1

3. Main Program

- 1) Get mathematical expression (키보드에서 산술식 입력) (ex: 2+4*3)
- 2) Build Tree (트리 생성 및 평가)
- 3) Do Tree Traversal (Inorder, Preorder, Postorder) (트리 운행결과 출력)

4. 상세 알고리즘 (강의노트 참조)

5. 검사절차

- 데이터 파일: 테스트 데이터 4개 → 트리 생성

Ex) 수식: 8+9-2*3, A / B * C * D + E, a+b*c-d

- 트리 순회 Output:

1) 데이터: (A / B * C * D + E)

Inorder: A / B * C * D + E Postorder: A B / C * D * E +

Inorder: + * * / A B C D E

2) 데이터: (8+9-2*3)

Inorder: 8+9-2*3 Postorder: 8 9 + 2 3 * - Preorder: - + 8 9 * 2 3

3) 데이터 : a+b*c-d: pre: +a-b-cd In: a+b*c-d post: ab+cd-*

4) 데이터: 3+2 →

- 만약 데이터가 숫자이면:

Eval Tree (알고리즘사용): Output: 11

