
프로그래밍언어 개론 보고서

[03] Tokenizer

제출일	2017/06/28
학 번	201000287
소 속	일어일문학과
이 름	유다훈

1 과제문제

1 주어진 코드에서 함수 구현

- Init_TM() : Transition Matrix 초기화
- nextToken() : 주어진 문자열을 검사하여 ID 혹은 INT 타입의 타입으로 분류

- 과제의 목적

- 특정 문자열을 입력 받을 경우, 해당 문자열을 나누어서 각 요소를 인식하여 속성 대로 분류하고 출력해주는 Scanner

2 과제 해결 방법

1 Init_TM()

- Transition Matrix 를 초기화해주는 동시에 문자열이 들어오면 어떠한 형태의 토큰인지를 검사하는 메소드
- 문자열의 첫 글자의 상태(ID 인지 INT 인지)에 따라 해당 문자열의 전체의 성격이 결정되므로, 2 차원배열로 되어있는 transM[][]의 행부분을 직접 지정해주고, 문자가 어떠한 형태인지에 따라 상태를 지정해주었다.

```
private void init_TM() {
    for(int i=0; i<4; i++)
        for(int j=0; j<128; j++)
            transM[i][j] = -1;

    transM[0]['-'] = 1; // '-'
    for(int i='0'; i<='9'; i++){
        transM[0][i] = 2;
        transM[1][i] = 2;
        transM[2][i] = 2;
        transM[3][i] = 3;
    }
    for(int i='A'; i<='Z'; i++){
        transM[0][i] = 3;
        transM[3][i] = 3;
    }
    for(int i='a'; i<='z'; i++){
        transM[0][i] = 3;
        transM[3][i] = 3;
    }

    //입력 받은 각각의 값을 확인하여 State 지정.
}
```

2 nextToken()

- 전달받은 문자열에서 문자를 하나씩 뽑아, 해당 문자에 해당하는 값을 배열의 열값에 지정해주고 그 값이 ID 인지, INT 인지 검사한다.
- 해당 값을 다시 행값으로 지정해주고 다음 문자값을 검사를 한다.
- 검사가 끝난 후에는 해당 문자열의 타입을 확인하여 문자열을 ID 혹은 INT 형태의 토큰으로 분류하고 저장한다.

```
private Token nextToken(){
    int StateOld = 0, StateNew;
    //토큰이 더 있는지 검사
    while(st.hasMoreTokens()) {

        //그 다음 토큰을 받음
        String temp = st.nextToken();
        Token result = null;

        //문자열의 문자를 하나씩 가져와 상태 판별
        for(int i = 0; i < temp.length(); i++)
        {
            StateNew = transM[StateOld][temp.charAt(i)];
            //입력된 문자의 상태가 reject이면 에러메세지 출력후 return함
            if(StateNew == -1)
            {
                System.out.println(String.format("acceptState error %s\n", temp));
                return null;
            }
            StateOld = StateNew;
        }

        //TokenType의 값을 확인하여서 state가 일치하는 경우 새로운 토큰을 생성하여 해당 값을 반환

        TokenType[] ts = TokenType.values();
        for( TokenType t : ts) {
            if(StateOld == t.finalState)
                result = new Token(t, temp);
        }

        return result;
    }
    return null;
}
```

3 과제를 해결하면서 느낀 점

- 1 Regular Expression 을 처음 접해보아서 init_TM() 메소드를 구현하는데 이해하기가 힘들었다.
- 2 평소 쓰던 for 문의 형태와는 다른 형태의 for 문을 사용해 볼 수 있었는데, 많이 편한 것 같다.
- 3 이번 과제에서 쓰인 enum 타입의 TokenType 이 이해가 어려웠다.
- 4 문자열을 문자 하나씩 검사를 실행하고 검사가 끝난 문자열을 어떻게 분류하는지에 관한 알고리즘을 구현해볼 수 있었다.