

PL Assignment #04 : Cute17 Scanner 보고서

컴퓨터공학과
201504280 신윤호

1. 문제 해결 방법

- token클래스의 `__init__`에서 토큰의 type과 lexeme을 초기화한다. type의 초기화는 lexeme이 keyword인지 아닌지에 따라 달라진다. 만약 `check_keyword` 함수의 반환값이 참 일 경우(lexeme이 keyword일 경우) `_get_keyword_type`함수를 이용해 입력된 keyword에 해당하는 type을 지정해준다. 그렇지 않을 경우(keyword가 아닌 경우) type 파라미터로 전달된 값으로 type을 지정해 준다. 그 후 `self.lexeme`에 파라미터로 전달된 lexeme 값을 대입해 lexeme을 지정해준다.
- transition matrix는 mDFA와 정규표현을 참조하여 작성하였다. 가령, 상태 0에서)가 입력되면 상태가 6으로 전이되므로 `self.transM[(0,')]'] = 6`으로 작성하였다.

2. 느낀점

지난번 과제를 수행하며 학습했던 오토마타 관련 내용들이 이번 과제에 많은 도움이 되었다. 이번 과제를 하며 지난번 과제로 배웠던 내용이 복습될 뿐만 아니라 더 심화해서 공부할 수 있었던 것 같다. 또한 매번 과제를 할 때 마다 코드를 읽으면서 파이썬 공부가 많이 되는 것 같다. 이번 과제를 수행하며 사전 자료형에 대해 더 많이 알게 되었고 eval함수, if~in구문, import등도 배울 수 있었다.

3. 테스트 코드 실행 결과

```
test_cute = CuteScanner("Test car + ' - * #T ( ) eq?")
test_tokens = test_cute.tokenize()
print test_tokens
```

```
201504280_hw04
D:\Python27\python.exe D:/대학/프로그래밍언어/과제/hw04/hw04/201504280_hw04.py
=== tokenize ===
Test car + ' - * #T ( ) eq?
scanning...
[ID: Test]
[CAR: car]
[PLUS: +]
[APOSTROPHE: ']
[MINUS: -]
[TIMES: *]
[TRUE: #T]
[L_PAREN: (]
[R_PAREN: )]
[EQ_Q: eq?]
[[ID: Test], [CAR: car], [PLUS: +], [APOSTROPHE: '], [MINUS: -], [TIMES: *], [TRUE: #T], [L_PAREN: (], [R_PAREN: )], [EQ_Q: eq?]]
Process finished with exit code 0
```