

컴파일러의 기초: Homework 1 for problem solving

C 에서의 Comment는 /* 으로 시작되고 */로 끝나며 nesting 이 허락되지 않는다. 따라서 C 의 Comment를 recognize하기 위해서는 위의 project 1 에서처럼 lex의 start condition feature를 이용할 수도 있지만 또 다른 방법은 그대로 regular expression 으로 표현하는 것이다. 이 문제에서는 regular expression을 이용하여 comment 를 인식하는 방식에 대해 다룬다.

(a) 정확히 C Comment 만을 accept 하는 DFA를 그리시오. 이 DFA가 예를 들어 /****/ 는 accept하고 /* */*/ 는 reject하는 지 확인하시오. 또한 두 세 개의 다른 까다로운 예에 대해서도 정확히 accept/reject 하는 지 확인하시오.

(b) 위의 DFA에 해당하는 C Comment의 regular expression을 구하시오.

(c) lexical analyzer에서 lookahead는 실제로 인식하려는 lexeme의 마지막 character 이후에 추가로 읽어야 하는 (이 lexeme을 인식하기 위해) character 의 개수를 의미한다. Comment를 이러한 방법으로 인식하는 C 컴파일러의 **maximum(worst-case) lookahead**는 infinite임을 보여주는 예를 하나 드시오(/ 와 * 는 C의 valid operator임을 기억할 것).

(d) 다음의 C 프로그램에 대해서 위의 두 방법 (start condition과 regular expression)을 사용한 C 컴파일러가 각각 어떤 behavior를 보일지 설명하시오.

```
main() {
    int a, b, c;
    a = b /* c;
}
```

※ (c)와 (d)는 실제로 lex programming을 통해 확인하는 것도 좋은 방법임!