

به نام خدا

تکلیف شماره 2

عنوان درس

کامپایلر پیشرفته

تهیه و تنظیم

محمد جواد فقیه نیا

ارائه شده به

جناب آقای دکتر پارسا

تاریخ تکمیل سند

1401/07/24

Convert line comment to block comment in C++ style:

دو روش برای این تمرین وجود دارد: 1- استفاده از Token.type -2 استفاده از Channel

نکته: در هر دو روش کامنت‌ها نباید skip شوند و باید آن‌ها را در کانال جداگانه‌ای نگهداری نمود. خط 44 فایل test.g4:

```
LINE_COMMENT: '//' ~ [\r\n]* -> channel(HIDDEN);
```

روش اول در فایل commentConverter.py خطوط 43 الی 46:

```
if lexer.symbolicNames[token.type - 2] == 'LINE_COMMENT':
    txt = token.text[2:]
    new_comment = "/* " + txt + " */\n"
    rewriter.replaceIndex(i, new_comment)
```

روش دوم: در این روش ابتدا whitespace را skip می‌نماییم تا تنها comment‌ها در کانال HIDDEN نگهداری شوند. خطوط 48 الی 52 در فایل commentConverter.py:

```
if token.channel == lexer.HIDDEN:
    if token.type == lexer.LINE_COMMENT:
        txt = token.text[2:]
        new_comment = "/* " + txt + " */\n"
        rewriter.replaceIndex(i, new_comment)
```

در صورتی که بخواهیم کانالی مخصوص کامنت‌ها تعریف نماییم باید lexer grammer و parser را در فایل‌ها جداگانه قرار دهیم در غیر اینصورت در فایل گرامر ترکیبی امکان تعریف کانال سفارشی وجود ندارد.

جهت اضافه نمودن هر متنی می‌توان مانند زیر کد را تغییر داد:

```
if lexer.symbolicNames[token.type - 2] == 'LINE_COMMENT':
    txt = token.text[2:]
    new_comment = "/* " + txt + " - Author: mohammad javad faghihniya */"
    rewriter.replaceIndex(i, new_comment)
```

نمونه‌ای از نتیجه در روش اول:

```
input string is (before):
    yx = a*b + bd * cc // salam
a = b // a
auy = (a * y) + n
a = 3 //num
a + n
(a + r) + r * 8 // test2 //test1 /* test
```

```
input string is (after):
    yx = a*b + bd * cc /* salam - Author: mohammad javad faghihniya */
a = b /* a - Author: mohammad javad faghihniya */
auy = (a * y) + n
a = 3 /* num - Author: mohammad javad faghihniya */
a + n
(a + r) + r * 8 /* test2 //test1 /* test - Author: mohammad javad faghihniya */
```