

Compiladores

Aluno: Lucas da Silva Lima

Professor: Christiano

Matrícula: 618031150

Observações:

O trabalho foi feito na versão 3.11 do PLY, uma vez que na versão 4.0 na assinatura da função yacc() da classe yacc já não existe mais a passagem de contexto 'method'.

Explicação do Código:

Para o léxico, temos:

```
tokens = ('ID', 'EQUALS', 'STAR')

t_ID = r'[a-zA-Z_][a-zA-Z0-9_]*'
t_EQUALS = r'='
t_STAR = r'\*'
t_ignore = " \t"

def t_error(t):
    print("Illegal character '%s'" % t.value[0])

lexer = lex.lex()
```

Foram criadas os tokens para ID que representa um alfanumérico, EQUALS para representar o símbolo '=' e STAR para representar o símbolo '*' da gramática do trabalho.

Os termos ignore e error são criados para fazer jus ao uso da ferramenta e por seguir os procedimentos de utilização indicados na documentação de boas práticas.

Para o sintático, temos:

```
def p_statement_assign(p):
    'S : L EQUALS R'

def p_statement_R(p):
    'S : R'

def p_L_START(p):
    'L : STAR R'

def p_L_ID(p):
    'L : ID'

def p_R_L(p):
    'R : L'

def p_error(p):
    if p:
```

```

        print("Syntax error at '%s'" % p.value)
    else:
        print("Syntax error at EOF")

parser = yacc.yacc(method='SLR', debug=True)

```

Foram criadas para cada termo da gramática um método.
No caso, estamos utilizando o method de parse tabela SLR.

Problema:

Ao executar o programa com o debug ativado e escolhendo o method 'SRL', podemos notar que será gerado por padrão o parse.out. E ao analisar o arquivo gerado vamos notar o seguinte texto:

```

48
49 state 2
50
51 (1) S -> L . EQUALS R
52 (5) R -> L .
53
54 ! shift/reduce conflict for EQUALS resolved as shift
55 EQUALS shift and go to state 6
56 $end reduce using rule 5 (R -> L .)
57
58 ! EQUALS [ reduce using rule 5 (R -> L .) ]
59
60

```

Executando o mesmo programa, porém trocando para method padrão (LR(1)), temos o seguinte:

```

55
56 state 2
57
58 (1) S -> L . EQUALS R
59 (5) R -> L .
60
61 EQUALS shift and go to state 8
62 $end reduce using rule 5 (R -> L .)
63
64

```

Visualmente já podemos ver uma diferença no 'state 2'. E, também temos um indicativo de conflito no terminal de comando informando que tivemos um conflito na execução do SRL.

A diferença se dá pelo fato de que utilizando o SRL teremos um conflito no termo "=", uma vez que duas actions serão geradas utilizando o termo, sendo elas uma action pelo SHIFT 6, e em seguida será reduzida pela produção 'R -> L .'. E sabendo que, no Follow de R define também uma action igual, contudo ele já foi reduzido e esperado uma ação de transferência e redução, contudo, não é o que ocorre, gerando um conflito de shift/reduce em torno do símbolo "=".

Contudo, isso já não ocorre no LR(1) uma vez que em seu algoritmo temos a implementação do lookahead, uma vez que esse recurso torna com que seja possível reconhecer os contexto à esquerda das produções geradas. E, sabendo disso, esse recurso é utilizado para resolver esses problemas de conflito shift/reduce.