



Universidad de Málaga
Departamento de Lenguajes y
Ciencias de la Computación
Campus de Teatinos, 29071 MÁLAGA

TRADUCTORES, COMPILADORES E INTÉRPRETES. EJERCICIOS YACC. TEMA 3. Curso 09/10

Nota: Los siguientes enunciados tienen por objetivo que **el alumno realice los programas Lex/YACC y que compruebe su funcionamiento correcto** mediante las pruebas que el alumno estime pertinentes. Al final de cada ejercicio se indica su dificultad y su extensión.

1.- Realizar los programas Lex/YACC correspondientes que reconozcan expresiones aritméticas expresadas en notación postfija, también llamada Notación Polaca Inversa (RPN- *Reverse Polish Notation*). Por ej.:

3 5 7 + * 10 -

que (en la notación normal o infija) equivale a $(3 * (5 + 7)) - 10$. Nótese que en RPN no es necesario el uso de paréntesis. **Fácil-Corto.**

2.- Partiendo de la gramática vista en clase:

```
e : e '+' t
   | t
   ;
t : t '*' f
   | f
   ;
f : NUM
   | ID
   | '(' e ')'
```

se pide ampliar dicha gramática para que acepte restas y divisiones. El programa YACC pedido no debe hacer uso de %left sino solucionar la ambigüedad en las propias reglas. Nótese que la suma y la resta tienen menos prioridad que el producto y la división. **Normal-Corto.**

2.- Dar una gramática recursiva por la izquierda (programas LEx/YACC) que permita reconocer sintácticamente una sentencia SELECT de SQL. Indicar si la gramática dada es ambigua o no. Los siguientes ejemplos de sentencias SELECT deben ser reconocidos por la gramática que se proponga:

- SELECT a FROM tabla1;
- SELECT a, 3, c FROM tabla1, tabla2 WHERE d = 7;
- SELECT alfa a, beta b, gamma g FROM tabla1, tabla2 WHERE d = 7 AND g = 8;

A efectos prácticos debe suponerse que:

- a) una expresión sólo puede estar formada por identificadores o números (ni siquiera hay expresiones aritméticas),
- b) los únicos operadores relacionales que hay son: la equivalencia, el mayor que y el menor que.
- c) se puede usar AND, OR y NOT.

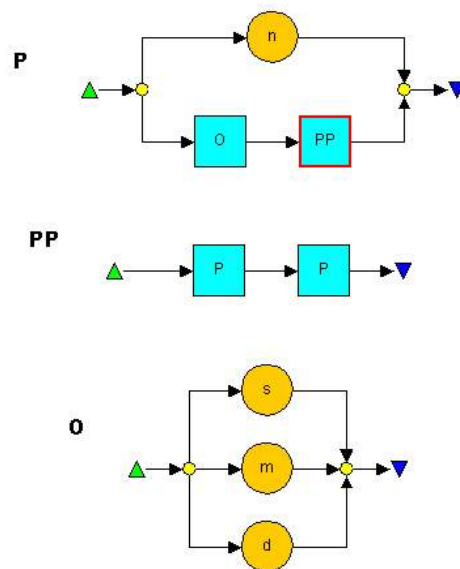
Complejo-Medio.

3.- Realizar los programas Lex/YACC correspondientes que permitan reconocer listas de números en las cuales también puede haber listas, ej.:

[12, 34, [56, 7, [5], 3, []], 3, [[[7, 8], 8], 9]]

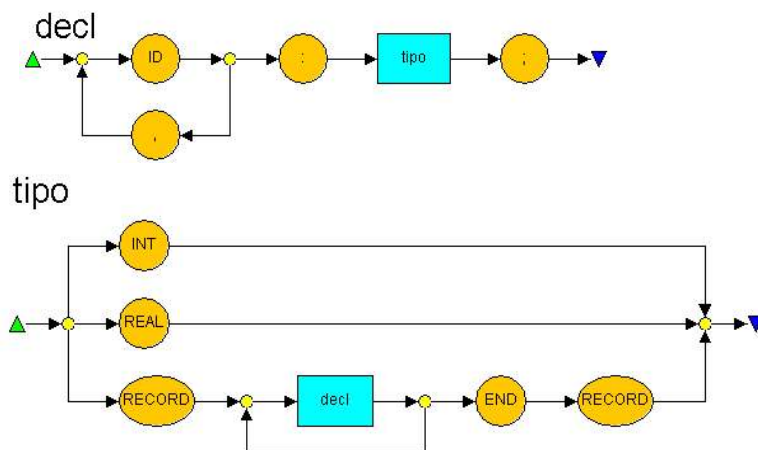
La lista vacía se considera válida. **Fácil-Corto.**

4.- Dados los siguientes diagramas de sintaxis:



se pide crear los programas Lex/YACC que reconozcan el mismo lenguaje. [Fácil-Corto.](#)

5.- Dados los diagramas de sintaxis:



se pide crear los programas Lex/YACC que reconozcan el mismo lenguaje. [Fácil-Corto.](#)

6.- Dar una gramática recursiva por la izquierda (programas Lex/YACC) que permita reconocer sintácticamente una sentencia SELECT de SQL. Indicar si la gramática dada es ambigua o no. Los siguientes ejemplos de sentencias SELECT deben ser reconocidos por la gramática que se proponga:

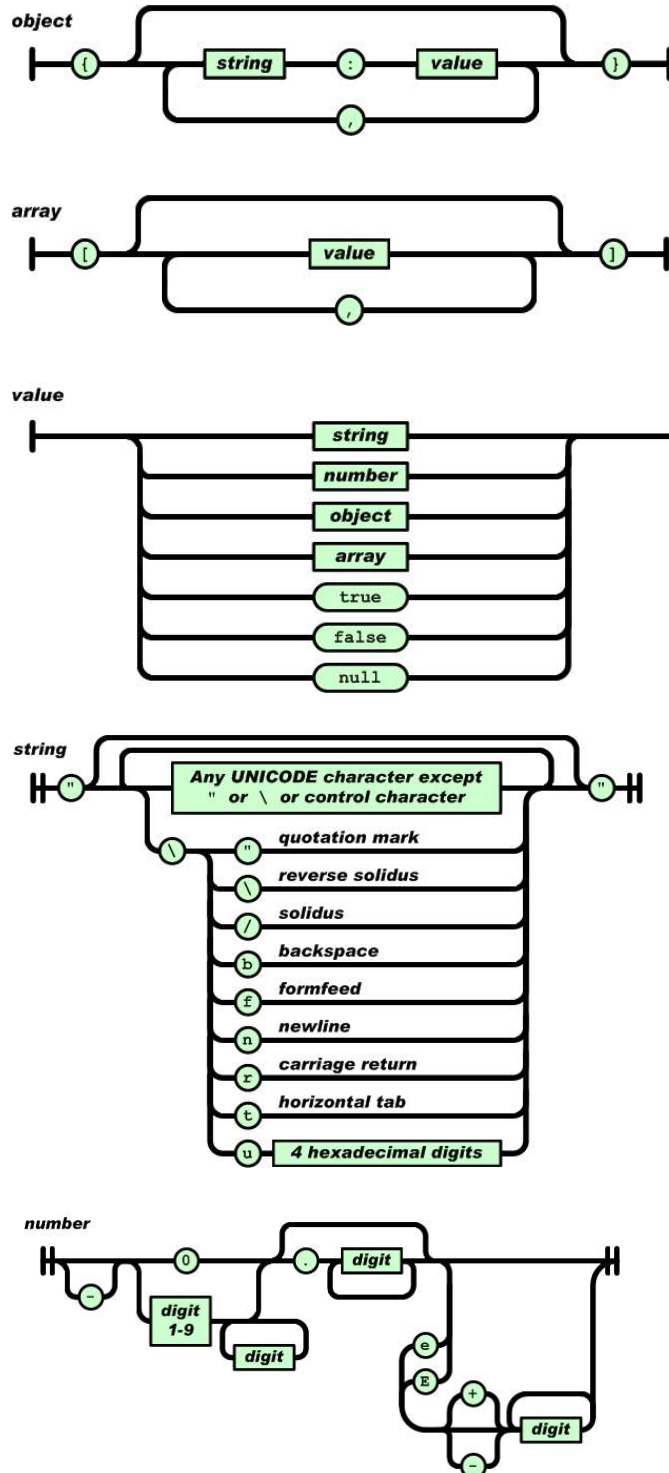
- SELECT a FROM tabla1;
- SELECT a, 3, c FROM tabla1 t1, tabla2 WHERE t1.d = 7;
- SELECT alfa a, beta b, gamma g FROM tabla1, tabla2 WHERE d = 7 AND g = 8;

A efectos prácticos debe suponerse que:

- a) una expresión sólo puede estar formada por identificadores o números (ni siquiera hay expresiones aritméticas),
- b) los únicos operadores relacionales que hay son: la equivalencia, el mayor que y el menor que.
- c) se puede usar AND, OR y NOT.

[Complejo-Medio.](#)

7.- Especificar los programas Lex/YACC necesarios para reconocer el lenguaje JSON que viene dado por los siguientes diagramas de sintaxis:



Se recomienda a l alumno que busque ejemplos en Internet. [Fácil-Largo](#).

8.- Crear los programas Lex/YACC necesarios para reconocer ficheros con extensión **.srt**. Estos ficheros se usan para contener subtítulos de películas:

<http://en.wikipedia.org/wiki/SubRip>.

Medio-Medio.