

Teoría de Lenguajes 2do. Parcial – Curso 2015

Consideraciones generales

- i) Escriba nombre y C.I. en todas las hojas.
- ii) Numere todas las hojas.
- iii) En la primera hoja, indique el total de hojas.
- iv) Comience cada ejercicio en una hoja nueva.
- v) Utilice las hojas de un solo lado.
- vi) Entregue los ejercicios en orden.

Ejercicio 1 [Evaluación individual del obligatorio]

a) Dada la siguiente EBNF que define una expresión (similar al laboratorio):

```
exp ::= expS ( opRel expS ) ?  
opRel ::= '=' | '<>' | '<' | '>' | '<=' | '>='  
expS ::= ( '+' | '-' ) ? term ( ( '+' | '-' | 'or' ) term ) *  
term ::= factor ( ( '*' | '/' | 'div' | 'mod' | 'and' ) factor ) *  
factor ::= 'id' | '( exp )' | 'not' factor
```

Indicar cuáles de los siguientes conjuntos de reglas para JCUP, sirven para generar un parser que reconozca el lenguaje definido por la EBNF anterior. Si no sirven, justificar por qué no.

i)

```
exp ::= expS | expS opRel expS;  
expS ::= expAux | PLUS expAux | MINUS expAux;  
expAux ::= term | term opA expAux;  
term ::= factor | factor opM term;
```

ii)

```
exp ::= expS | expS opRel expS;  
expS ::= term | term opA expS  
| PLUS term | PLUS term opA expS  
| MINUS term | MINUS term opA expS;  
term ::= factor | factor opM term;
```

iii)

```
exp ::= expS | expS opRel expS;  
expS ::= opX expAux;  
opX ::= PLUS | MINUS | ;  
expAux ::= term | term opA expAux;  
term ::= factor | factor opM term;
```

iv)

```
exp ::= expS | expS opRel expS;  
expS ::= term | term opA term  
| PLUS term | PLUS term opA term  
| MINUS term | MINUS term opA term;  
term ::= factor | factor opM factor;
```

donde para todos caso se definen: factor, opRel, opA y opM de la siguiente forma:

```
factor ::= ID | LPAREN exp RPAREN | NOT factor;  
opRel ::= EQUAL | NOTEQUAL | LESS | GREATER | LESSOREQUAL | GREATEROREQUAL;  
opA ::= PLUS | MINUS | OR;  
opM ::= ASTERISK | SLASH | DIV | MOD | AND;
```

b) ¿En el laboratorio cuando una tira (o entrada) a reconocer no pertenece al lenguaje especificado en JCUP, qué sucede? Explique brevemente como se corrige.

- i)** Devuelve error sintactico
- ii)** Se produce un error de compilación
- iii)** No genera el parser
- iv)** Devuelve un conflicto shift/reduce o reduce/reduce

c) Dadas las siguientes reglas para JCUP:

```

props ::= props SEMICOLON prop | prop;
prop  ::= ID ASSIGN exp
        { : System.out.println("ASIGNACION"); : }
        | REPEAT props UNTIL exp
        { : System.out.println("REPEAT"); : }
        | IF exp THEN prop
        { : System.out.println("IF"); : };
    
```

y el siguiente texto de entrada:

```

exito := 0; d := 1;
repeat
  d := d+1
until (d * d > n) or (n mod d = 0);
if (n mod d = 0) then exito := 1
    
```

Indicar cuál es la salida que producirá el parser al procesar el texto de entrada.

| i) | ii) | iii) | iv) | v) |
|------------|------------|------------|------------|------------|
| ASIGNACION | ASIGNACION | ASIGNACION | ASIGNACION | ASIGNACION |
| IF | REPEAT | ASIGNACION | ASIGNACION | ASIGNACION |
| ASIGNACION | IF | ASIGNACION | REPEAT | ASIGNACION |
| REPEAT | | REPEAT | ASIGNACION | ASIGNACION |
| ASIGNACION | | ASIGNACION | IF | REPEAT |
| ASIGNACION | | IF | ASIGNACION | IF |

Ejercicio 2 [30 puntos]

Dado el siguiente lenguaje:

$$L_2 = \{ \text{tiras de la forma } w_1w_2\dots w_N\#x_1x_2\dots x_N\#w_1x_1w_2x_2\dots w_Nx_N ; w_i, x_i \in \{0,1\} \}$$

- Clasifique L_2 según la Jerarquía de Chomsky. Justifique.
- Construya una gramática $G_2 / L_2 = L(G_2)$.
- Construya un autómata $M_2 / L_2 = L(M_2)$.

Nota: la tira $\#\#\notin L_2$

Ejercicio 3 [20 puntos]

Dados los siguientes lenguajes:

$$L_a = \{ x / x \in \{0,1\}^* \text{ donde } x \text{ es de la forma } (01)^p0^t1^m \text{ con } p,m > 0 ; t \geq 0 \}$$

$$L_b = \{ x / x \in \{0,1\}^* \text{ donde } x \text{ es de la forma } (01)^p0^t1^m \text{ con } p > m > 0 ; t \geq 0 \}$$

- Construya gramáticas simplificadas G_a y $G_b / L_a = L(G_a)$ y $L_b = L(G_b)$. Deberá justificar porque las gramáticas propuestas están simplificadas.
- Construya un autómata $M_b / L_b = L(M_b)$. ¿Es determinista? Justifique. Si fuera NO determinista, indique todos los casos donde se produce dicho NO determinismo.

Ejercicio 4 [4 puntos]

Decir si las siguientes afirmaciones son Verdaderas o Falsas. Justifique.

- Si L_1 es recursivamente enumerable, entonces es infinito
- Si L_2 es recursivamente enumerable pero NO libre de contexto y L_3 es libre de contexto pero NO regular, entonces $L_4 = L_2 \cap L_3 = \emptyset$

Nota: Las gramáticas y los autómatas deben corresponderse con el tipo del lenguaje considerado en cada caso, según la Jerarquía de Chomsky. Se valora positivamente la simplicidad de las soluciones propuestas así como una breve explicación de éstas. Todas las respuestas deben estar debidamente justificadas.