Teoría de Lenguajes 2do. Parcial – Curso 2017

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) i) ¿En BNF qué representan []?
ii) ¿En BNF cómo se representan los terminales?
iii) ¿Cómo se representan los terminales en las Gramáticas Libres de Contexto vistas en el laboratorio?

b) Dada la siguiente EBNF que define un lista de parámetros (similar a la lista de parámetros de la declaración de funciones del laboratorio):

```
type ::= simpleType | arrayType | "identif"  
simpleType ::= constant ".." constant | "(" identifList ")"  
arrayType ::= "array" "[" simpleType ( "," simpleType ) * "]" "of" type  
identifList ::= "identif" ( "," "identif" ) *  
constant ::= ("+" | "-" ) ? ("identif" | "unsignednumber")
```

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

i)

type	->	simpleType arrayType 'IDENT'
simpleType	->	constant 'DOTDOT' constant 'OPAR' identList 'CPAR'
arrayType	->	'ARRAY' 'OSQRBRA' simpleType arrayType2
arrayType2	->	'COMMA' simpleType arrayType2 'CSQRBRA' 'OF' type
identList	->	'IDENT' 'COMMA' 'IDENT'
constant	->	'IDENT' 'ADD' 'IDENT' 'SUB' 'IDENT' 'NUMBER' 'ADD' 'NUMBER' 'SUB' 'NUMBER'

ii)

type	->	simpleType arrayType 'IDENT'
simpleType	->	constant 'DOTDOT' constant 'OPAR' identList 'CPAR'
arrayType	->	'ARRAY' 'OSQRBRA' simpleType arrayType2
arrayType2	->	'COMMA' simpleType arrayType2 'CSQRBRA' 'OF' type
identList	->	'IDENT' 'COMMA' identList 'IDENT'
constant	->	'IDENT' 'ADD' 'IDENT' 'SUB' 'IDENT' 'NUMBER' 'ADD' 'NUMBER' 'SUB' 'NUMBER'

iii)

type	->	simpleType arrayType 'IDENT'
simpleType	->	constant 'DOTDOT' constant 'OPAR' identList 'CPAR'
arrayType	->	'ARRAY' 'OSQRBRA' simpleType CSQRBRA' 'OF' type
identList	->	'IDENT' 'COMMA' identList 'IDENT'
constant	->	'ADD' 'IDENT' 'SUB' 'IDENT' 'ADD' 'NUMBER' 'SUB' 'NUMBER'

Ejercicio 2 [8 puntos]

Sean los siguientes lenguajes sobre un mismo alfabeto Σ . Indique si las siguientes afirmaciones son verdaderas o falsas, justificando en cada caso.

- L_1 Libre de Contexto no Regular y L_2 Libre de Contexto no vacío. Entonces $L_1 \cap L_2$ no es Regular.
- L_3 Recursivamente Enumerable no Libre de Contexto y L_4 Libre de Contexto no Regular. Entonces $L_3 \cap L_4$ es Libre de Contexto.
- L_5 Finito (no vacío) y L_6 Recursivamente Enumerable no Libre de Contexto. Entonces $L_5 \cap L_6$ es Libre de Contexto no Regular.
- $L_7 \cap L_8$ es Libre de Contexto no vacío. Entonces L_7 y L_8 son ambos Libres de Contexto.

Ejercicio 3 [19 puntos]

Dados los siguientes lenguajes:

$$L_a = \{ 0^k 1^{j+1} 2^t 0^r, \text{ donde } r > 0, j \text{ par}, k, t \geq 1 \}$$
$$L_b = \{ 0^k 1^{j+1} 2^t 0^r, \text{ donde } r = j/2, j \text{ par}, k, t \geq 1 \}$$

- 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.

Ejercicio 4 [28 puntos]

Sea

$$L_4 = \{ x \# y / x, y \in \{a,b\}^* \text{ donde } y \in \text{Perm}(x) \}$$

Perm(x) es el conjunto de tiras que tienen exactamente los símbolos de x pero no necesariamente en el mismo orden.
Por ejemplo, $\text{Perm}(aba) = \{aba, aab, baa\}$

- Clasifique L_4 según la Jerarquía de Chomsky.
- Construya una gramática G_4 / $L_4 = L(G_4)$.
- Construya un autómata M_4 / $L_4 = L(M_4)$.