

Procesadores de Lenguaje. Curso 2000-2001. Febrero de 2001. Cuestiones.

No se permiten libros ni apuntes. Se tendrá que superar la parte de cuestiones y de problemas para aprobar la asignatura. Tiempo: 1 hora 15 min.

1. (2.5 puntos) Dada una gramática con una producción del tipo $A \rightarrow \alpha_1 | \dots | \alpha_n$ con $\alpha_i \in (V_T UV_N)^+$ para todo i . ¿Qué condición han de cumplir las funciones $PRIM(\alpha_i)$ para que la gramática pueda ser analizada mediante el método de descenso recursivo predictivo?. En base a ello justifica si el lenguaje generado por la siguiente gramática puede ser reconocido mediante este método.

$$A \rightarrow u | Av$$

con $u, v \in V_T$. En caso negativo modifica la gramática para que pueda ser analizada por dicho método.

2. (2.5 puntos) Supongamos la siguiente gramática para generar números enteros binarios, diseña un ETDS para traducir un número binario entero a un número decimal entero.

$$N \rightarrow NB | B$$

$$B \rightarrow 1 | 0$$

Construye el árbol de análisis sintáctico para la cadena 1101 y ejecuta las reglas semánticas para encontrar su valor decimal. Indica el orden de evaluación de los atributos.

3. (2.5 puntos) Supongamos una gramática de atributos sintetizados, justifica si el método de análisis sintáctico descendente permite realizar al mismo tiempo el análisis sintáctico y la evaluación de los atributos en el análisis semántico. ¿Qué dirías entonces respecto al caso en que el método es ascendente y los atributos heredados?
4. (2.5 puntos) Dada la siguiente gramática

$$E \rightarrow (E) | n$$

Diseña un analizador LR para reconocer el lenguaje generado por dicha gramática e incorpora el método de recuperación de errores mediante producciones de error. ¿Es LR(0)? ¿Es SLR(1)? Indica el contenido de la pila y la cadena de entrada al analizar la cadena (n).

Procesadores de Lenguaje. Curso 2000-2001. Febrero de 2001. Problemas.

Se permiten libros y apuntes. Se tendrá que superar la parte de cuestiones y de problemas para aprobar la asignatura. Tiempo: 1 hora 30 min.

1. (5 puntos) Dada la siguiente gramática de operadores que genera expresiones lógicas

$$E \rightarrow E \wedge E \mid E \vee E \mid \neg E \mid E \iff E \mid id$$

y cuya precedencia de operadores es de menor a mayor \iff , \vee , \wedge , \neg . Se pide:

- (a) Construir la tabla de precedencia de operadores, suponiendo que los operadores \wedge , \vee , \iff son asociativos a la izquierda y el operador \neg asociativo a la derecha.
 - (b) Analizar la cadena de entrada $a \vee b \wedge \neg c$ mediante el algoritmo de análisis sintáctico por precedencia de operadores. Dibuja el árbol de derivación.
2. (5 puntos) La siguiente gramática genera un sencillo lenguaje con expresiones aritméticas, declaraciones de variables e instrucciones de asignación y de entrada y salida.

$$programa \rightarrow lista_decl \ lista_sent$$

$$lista_decl \rightarrow decl \ ; \ lista_decl \mid \epsilon$$

$$decl \rightarrow \mathbf{var} \ id$$

$$lista_sent \rightarrow sent \ ; \ lista_sent \mid \epsilon$$

$$sent \rightarrow id \ := \ E \mid \mathbf{leer} \ id \mid \mathbf{escribir} \ id$$

$$E \rightarrow E + E \mid E \uparrow E \mid id \mid num$$

Añade las reglas semánticas y los atributos necesarios para:

- (a) Obtener el número de variables declaradas y una lista con el nombre de cada variable.
- (b) Obtener el valor numérico de las expresiones.
- (c) Construye el árbol de análisis sintáctico para la siguiente entrada indicando las dependencias entre los atributos y un posible ordenamiento de los nodos del árbol en el grafo de dependencias.

```
var a;
var b;
leer a;
b= a ↑ a + a ;
escribir b
```

Suponed que el operador potencia tiene mayor precedencia que el operador de adición y es asociativo a la derecha.