

Práctica. Segunda Fase

Análisis sintáctico descendente

Los objetivos de esta segunda fase son:

- Especificar la sintaxis del lenguaje descrito en la primera fase.
- Diseñar e implementar manualmente un analizador sintáctico predictivo recursivo.
- Generar automáticamente un analizador sintáctico predictivo recursivo con JavaCC

Los productos relativos a esta fase son los siguientes:

- (1) Una memoria con los siguientes sub-apartados:
 - a) La especificación sistemática de la sintaxis mediante una gramática incontextual. Para ello, deberán aplicarse los patrones sistemáticos discutidos en teoría. Así mismo, en lo referente a la sintaxis de las expresiones, deben considerarse las siguientes prioridades y asociatividades de los operadores contemplados (0 es el nivel de menor prioridad):
 - Nivel 0: + asocia a derechas, - (binario) no asocia.
 - Nivel 1: **and** y **or** asocian a izquierdas.
 - Nivel 2: Los operadores relacionales. Todos ellos son operadores binarios, infijos, asociativos a izquierdas.
 - Nivel 3: *, /. Operadores binarios, infijos, no asociativos.
 - Nivel 4: - (unario) y **not**. Operadores unarios, prefijos. - no asocia, **not** asocia.
 - b) Transformaciones necesarias para obtener una gramática LL(1) equivalente, que permita el análisis descendente.
 - c) *Primeros* y *siguientes* de cada no terminal, y *directores* de cada regla en la gramática transformada (sugerencia: utiliza una herramienta online, como Proletool, <http://portal.esi.uclm.es/proletool/> como ayuda para obtener estos símbolos).
- (2) Una implementación manual en Java del analizador sintáctico descendente predictivo recursivo. Dicha implementación debe utilizar el analizador léxico generado con JLex en la primera fase.
- (3) Una implementación con el generador de analizadores descendentes predictivos recursivos JavaCC

Entrega:

Fecha límite de entrega: **Domingo 8 de marzo de 2020, a las 23:59 pm.**

Modo de entrega: A través del campus virtual, en un único .zip. Dicho archivo debe contener: (i) un documento PDF memoria.pdf con la memoria; (ii) una carpeta implementación_manual, en el interior de la cuál debe incluirse la implementación manual del procesador (analizador léxico, analizador sintáctico, módulo de gestión de errores); (iii) una carpeta implementación_javacc, en el interior de la cuál debe incluirse la implementación 'javacc'; (iv) una carpeta pruebas con distintos programas de prueba que permiten probar el procesador. La entrega debe ser realizada solamente por un miembro del grupo.