**Trabajo Práctico No2**

**Teoría de Lenguajes y Autómatas**

**Integrantes:**

**Homovc, Federico**

**Reznik, Luciana**

**Pereyra, Cristian**

**1er cuatrimestre, año 2012**

# Objetivo del Trabajo Práctico

El trabajo consiste en programar en C un Generador de Analizador Sintáctico Descendente con Retroceso(ASDR). Dada una gramática libre de contexto, sin recursividad a izquierda, el programa deberá generar un ASDR que estará escrito en lenguaje C.(ASDR.c). El ASDR podrá luego recibir una cadena de entrada y determinar si pertenece o no al lenguaje generado por la gramática, mostrando además la derivación que llevó a dicha cadena.

# Consideraciones realizadas

Para hacer el trabajo práctico se buscó centrarse en lo visto en las clases teóricas de la materia, y se trató de realizar la programación con los algoritmos vistos en las prácticas.

Nos basamos en el trabajo práctico anterior, al cual le agregamos algunas ligeras modificaciones para poder soportar gramáticas libres de contexto, a diferencia del anterior que soportaba solo gramáticas regulares.

# Descripción del desarrollo del TP

Para la realización del trabajo, se utilizaron las librerías de la materia Sistemas Operativos, para facilitar el parseo de las estructuras de autómata y gramática. Utilizando listas, mapas y conjuntos, la lógica de las implementaciones se simplificó bastante y se redujo a un nivel más comprensible y legible.

Nos basamos también en el TP Nº1 de la materia de donde sacamos el parser, las estructuras de datos de las gramáticas, producciones, y demás.

# Dificultades encontradas en el desarrollo del TP

La mayor dificultad a la hora de desarrollar el trabajo práctico vino de la mano del generador de código C, el cual se tornó dificultoso debido a que teníamos problemas a la hora de hacer los *retrocesos*, pero decidimos implementar como se sugiere, una heurística basada en la longitud de la cadena, a partir de la cual el algoritmo funcionó adecuadamente.

# Futuras extensiones

La mayor dificultad a la hora de desarrollar el trabajo práctico vino de la mano del generador de código C, el cual se tornó dificultoso debido a que teníamos problemas a la hora de hacer los *retrocesos*, pero decidimos implementar como se sugiere, una heurística basada en la longitud de la cadena, a partir de la cual el algoritmo funcionó adecuadamente.