

**Task 1.** En este taller vamos a construir un reconocedor (lexer y parser), utilizando JavaCC, para un lenguaje sencillo. El lenguaje a utilizar es Logo, un lenguaje sencillo e intuitivo para aprender a programar. Logo tiene conceptos básicos de movimiento de una tortuga en una cuadrícula. El objetivo de la tortuga es pintar figuras a medida que se va moviendo.

1. Ingrese a <https://turtleacademy.com> para conocer la implementación de Logo en que se basa este taller. Basta que consulten la sección *Lessons*, la información que necesitan se encuentra entre **1. Logo's turtle** y **14. Recursión**. El fragmento de Logo que debe reconocer su parser consiste de:

- Los comandos básicos (forward, right, penup,...).
- Repeat loops.
- Declaración de variables.
- Definición de funciones (*procedures*) con o sin parámetros.
- Llamado de funciones. Esto implica recursión: que una función llame a otra, posiblemente ella misma.

Tenga en cuenta lo siguiente:<sup>1</sup>

- Como separador de instrucciones pueden usar *punto* “.” o *newLine* “\n”.
  - No hace falta que una función haya sido declarada para llamarla.
  - No importa que se sobrelapen nombres de funciones, parámetros y variables.
  - No importa que un programa no termine, esto no es un problema de compilación.
2. Declare la gramática que va a implementar. Puede ser como un comentario de java en el archivo .jj del parser, o puede agregarla a un pdf de documentación, o puede ser a mano y escaneada en un archivo aparte, etc. No hace falta que usen BNF, pueden usar convenciones de expresiones regulares como las que admite javaCC para hacer más compacta su gramática.
  3. Implemente el parser en javaCC.
  4. Suba a Bloque Neón únicamente el archivo .jj

---

<sup>1</sup>Entre otras razones, porque sólo nos interesa el aspecto independiente del contexto