

Task 1. Para este taller vamos a construir un Parser en GOLD para el lenguaje Logo. Puede asumir que el programa ha sido previamente tokenizado, de modo que cada instrucción se representa con un único caracter: 'f' (FORWARD), 'r' (RIGHT), 'e' (REPEAT), 'n' (number), etc...

Recuerde las instrucciones básicas de Logo:

1. **FORWARD x:** Mueve a la tortuga hacia adelante x posiciones. Alternativamente, este comando se puede utilizar en su forma comprimida **FD x**.
2. **RIGHT x:** Gira a la tortuga sobre su eje x grados con las manecillas del reloj. Alternativamente, este comando se puede utilizar en su forma comprimida **RT x**.
3. **LEFT x:** Gira a la tortuga sobre su eje x grados contra las manecillas del reloj. Alternativamente, este comando se puede utilizar en su forma comprimida **LT x**.
4. **REPEAT x [Y]:** Se utiliza para representar ciclos, donde las instrucciones entre los paréntesis cuadrados se repiten x veces.
5. **PENUP:** Se utiliza para levantar el lápiz de la tortuga (evitando que raises the turtles pen (causing it not to draw)
6. **PENDOWN:** Which puts down the turtles pen (causing it to draw)

Cada instrucción esta delimitada por un punto . (denotando el final de la instrucción).

- Defina tokens de un solo caracter para cada instrucción de Logo, y un token para los números decimales.
- Defina una gramática para los programas de Logo. El conjunto de símbolos terminales debe incluir los tokens definidos en el paso anterior, junto con los demás elementos de sintáxis: '[', ']', '.'
- Implemente en GOLD un autómatas de pila que reconozca el lenguaje de esta gramática, puede seguir el esquema visto en clase.

Programas ejemplo.

```
1 PENDOWN.  
2 FORWARD 10.  
3 RIGHT 90.  
4 FORWARD 10.  
5 RIGHT 90.  
6 FORWARD 10.  
7 RIGHT 90.  
8 FORWARD 10.
```

Snippet 1: Square

```
1 PENDOWN.  
2 REPEAT 4 [  
3 FORWARD 10.  
4 RIGHT 90.  
5 ] .
```

Snippet 2: Square

```
1 REPEAT 3  
2 [PENDOWN.  
3 REPEAT 2 [FD 40.  
4 RT 135.  
5 FD 50.  
6 LT 135.] .  
7 RIGHT 90.  
8 PENUP.  
9 FORWARD 20.  
10 LEFT 90.  
11 ] .
```

Snippet 3: 3 Mountains