

## PRÁCTICA 2: Programar un simulador de una Máquina de Turing

### Objetivo

El objetivo de la práctica consiste en programar un simulador de una máquina de turing, realizando un diseño orientado a objetos.

### Entrega

La ejecución de la práctica será revisada en la sesión de entrega en el laboratorio.

En la tarea del aula virtual se debe entregar el código fuente del simulador y las definiciones de las Máquinas de Turing ( $M = (Q, \Sigma, \Gamma, s, b, F, \delta)$ ) diseñadas para resolver los dos problemas propuestos en el enunciado.

Fecha límite de entrega: 8-9 de noviembre de 2018.

### Notas de implementación

Posibles variaciones de la Máquina de Turing a implementar:

- Máquina de Turing con escritura y movimientos simultáneos o independientes.
- Máquina de Turing donde los únicos movimientos sean izquierda (L) y derecha (R) o que incluya también la posibilidad de no movimiento (S).
- Máquina de Turing con cinta infinita en una única dirección o en ambas direcciones.

Los elementos de la Máquina de Turing se introducirán en tiempo de ejecución del programa utilizando un fichero de texto con el siguiente formato:

# Comentarios

q1 q2 q3 ...           # conjunto Q

a1 a2 a3 ...           # conjunto  $\Sigma$

A1 A2 A3 ...           # conjunto  $\Gamma$

q1                      # estado inicial

b                       # símbolo blanco

q2 q3                   # conjunto F

q1 a1 q2 a2 m          # función de transición:  $\delta(q1, a1) = (q2, a2, m)$

...                      # cada una de las transiciones en una línea distinta

Nota: La estructura anterior de las transiciones puede modificarse según las características de la Máquina de Turing a implementar.

Se debe verificar que la información proporcionada en el fichero se corresponda con la definición correcta de una Máquina de Turing..

Para una ejecución con una Máquina de Turing determinada, el simulador debe poder ejecutarse con diferentes parámetros de entrada, que serán introducidos por teclado o por fichero.

Inicialmente, la cabeza de L/E debe encontrarse en el primer símbolo de los parámetros de entrada.

Como salida, el programa debe indicar si la Máquina de Turing se ha parado en un estado de aceptación y mostrar la cinta hacia la derecha de la cabeza de L/E hasta encontrar el primer símbolo blanco después del resultado.

Se puede realizar el código utilizando los lenguajes C++ o Java.

Se debe utilizar un diseño orientado a objetos.

De forma opcional, el programa podrá aceptar Máquinas de Turing Multicintas.

### **Máquinas de Turing a diseñar:**

1. MT que reconozca el lenguaje  $L = \{ w \in \Sigma^* \mid w = (a \mid b \mid c)^* \wedge \text{la combinación } ab \text{ se encuentra un número par de veces} \}$
2. MT que reciba como parámetro una cadena binaria y copie en la cinta de entrada primero los unos y después los ceros, dejando un símbolo blanco en medio de cada bloque.

Ejemplo: Cadena de entrada: •100101100•

Resultado final: •100101100•1111•00000•