

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

El desarrollo de la práctica será revisado en las sesiones de 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: 14 de noviembre de 2020.

Notas de implementación

Posibles variaciones de la Máquina de Turing a implementar: (seleccionar una opción en cada fila)

- 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 cumpla con las restricciones de la definición formal de una Máquina de Turing, por ejemplo: $s \in Q$.

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.

Una vez cargada la definición de la Máquina de Turing, debe ser posible ejecutarla con diferentes parámetros de entrada. Los parámetros para las diferentes ejecuciones podrán ser introducidas 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 adicional no obligatoria, el programa podrá aceptar Máquinas de Turing Multicintas.

Máquinas de Turing a diseñar:

1. MT que reconozca el lenguaje $L = \{ a^n b^m \mid m > n, n > 0 \}$
2. MT que reciba como parámetro una cadena binaria y elimine los ceros de la palabra dejando únicamente la secuencia de unos sin espacios en blanco:

Ejemplo: Cinta con la cadena de entrada: •100101100•

Cinta con el resultado final: •1111•