

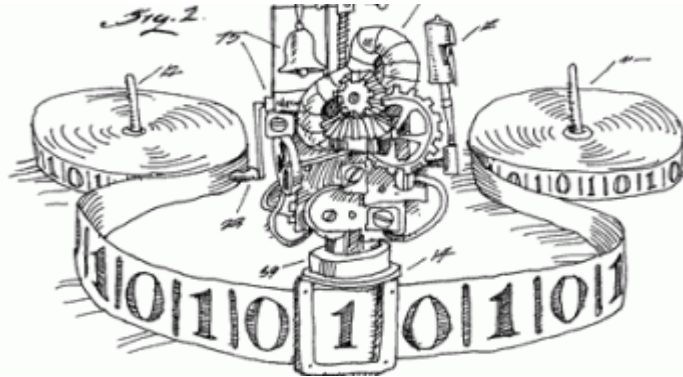


# **Universidad Austral de Chile**

**Facultad de Ciencias de la Ingeniería  
Escuela de Ingeniería Civil en Informática**

## ***Simulador de Máquina de Turing***

***“Lenguaje y estructuras de datos a utilizar”***



**Integrantes: Felipe Aguilera G.  
Marcelo Arriagada S.  
Rodrigo Miranda M.**

**Profesora: María Eliana de la Maza W.**

**Asignatura: Teoría de Autómatas**

**Informe: N°2**

**Fecha de entrega: 06 de junio, 2017**

## **Introducción**

En este informe se definirán las estructuras de datos y el lenguaje de programación que se utilizarán para el desarrollo de nuestro programa.

Para esto, se dará cierta estructura para las transiciones de la máquina, estados inicial y final; y por último la palabra de entrada.

# Desarrollo

## 1.- Lenguaje de programación: C++

**1.1.- Motivo:** En nuestra experiencia como estudiantes, estamos más familiarizados con este lenguaje.

## 2.- Transiciones de la máquina: Arreglo bidimensional de registros

**2.1.- Motivo:** Como se ha visto en clases, al momento de realizar una Máquina de Turing, recurrimos en primera instancia, a una tabla. La tabla estará conformada por los datos que se otorgarán por el usuario, al momento de ingresar los estados inicial y final; que corresponderán a las filas. Las columnas estarán dadas por las transiciones de la máquina, en específico, se extraerá la información de lo que ocurre en ese estado al momento de leer ese símbolo; y eso nos entregará las columnas. En cada celda, tendremos 3 tipos de información: el estado, la acción a realizar en la cinta y la dirección en la que nos moveremos; considerado como registro.

## 3.- Estados inicial y final: Variables enteras.

**3.1.- Motivo:** La primera razón es porque estos datos nos entregarán la dimensión de filas que tendrá la tabla (arreglo). Y en segundo lugar, para realizar las transiciones de la máquina (moverse de izquierda a derecha o viceversa) simplemente sumamos o restamos estas variables, situándonos de esta forma, en el estado correspondiente.

## 4.- Palabra de entrada: lista doblemente enlazada

**4.1.- Motivo:** Al utilizar este tipo de estructura, nos da la facilidad de movernos en una dirección bidireccional por cada nodo (cada símbolo de la palabra en la cinta).