

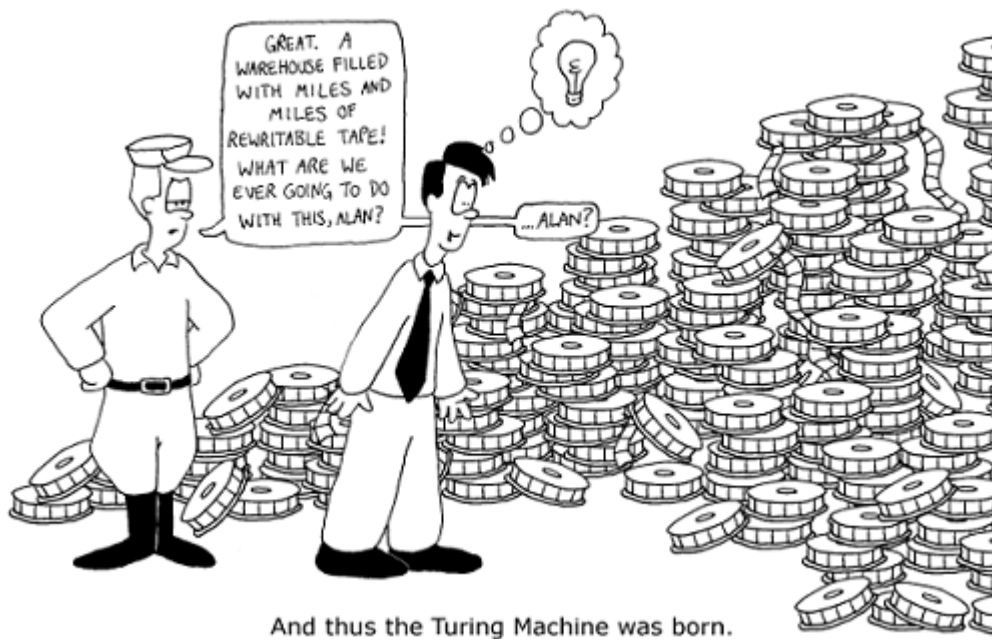
## Problema F: Turing

Base name: turing

Autor: Juan Manuel Reyes

Revisó: Javier Torres

Tiempo Límite: 1 segundo



Una máquina de Turing es un dispositivo que manipula símbolos sobre una tira de cinta de acuerdo con una tabla de reglas. A pesar de su simplicidad, una máquina de Turing puede ser adaptada para simular la lógica de cualquier algoritmo de computador y es particularmente útil en la explicación de las funciones de la unidad central de procesamiento de un computador.<sup>1</sup>

El científico de la computación Sédna Labázitsira acaba de

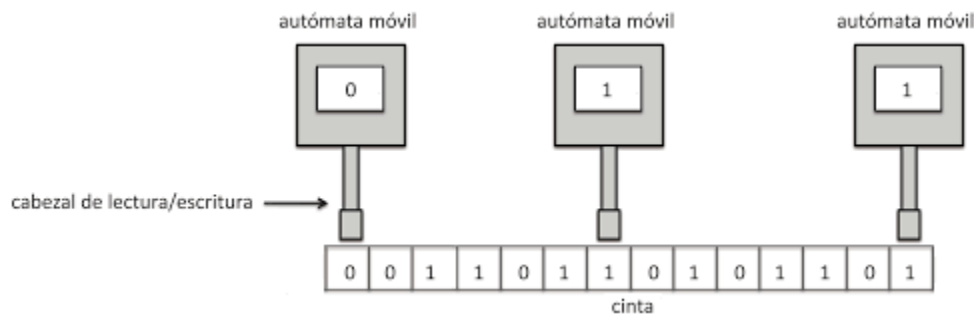
proponer una máquina de Turing finita redimensionable pero de tamaño no limitado con tres cabezas. Es muy importante para la continuidad de su investigación probar su funcionamiento y por tal razón le ha pedido que dada una serie de instrucciones que serán definidas a continuación, usted simule la máquina y ejecute cada una de ellas.

Tenga en cuenta que las posiciones de la lista son 0-índice, es decir, los índices empiezan en 0 y por tanto la primera posición es 0, la segunda posición es 1, y así sucesivamente.

<sup>1</sup> [https://es.wikipedia.org/wiki/M%C3%A1quina\\_de\\_Turing](https://es.wikipedia.org/wiki/M%C3%A1quina_de_Turing)

## Laboratorio sobre Listas Enlazadas, basado en un problema de la Maratón Interna de Algoritmos

### Categoría: Turing



La máquina tiene 3 cabezas lectoras: C0, C1 y C2, que en todo momento se encuentran al inicio de la cinta (C0), es decir sobre el primer elemento de la cinta, en la mitad de la cinta (C1), es decir, sobre el elemento de la mitad de la cinta (sobre el  $n/2$ -ésimo elemento, si la cinta tiene  $n$  elementos), y al final de la cinta (C2), es decir, sobre el último elemento.

Las cabezas pueden llevar a cabo tres operaciones:

1. Leer el elemento sobre el cual se encuentran.
2. Agregar un elemento. Que en cada caso implica dejar el nuevo elemento en la misma posición en la que cada cabeza quedará. Si se inserta por C0, el nuevo elemento estará de primero, si se inserta por C1, quedará en la mitad y si se inserta por C2 quedará de último.
3. Eliminar un elemento. Se elimina el elemento sobre el cual está la cabeza que está eliminando el valor.

Las tres operaciones se podrán hacer en cualquier momento. La cinta de la máquina inicialmente estará vacía y por tanto, las tres cabezas lectoras se encontrarán en el mismo lugar, sobre ningún elemento.

Si se lee un elemento sobre una cabeza lectora que se encuentra sobre ningún elemento, el valor leído será #. Si se elimina un elemento sobre una cabeza lectora que se encuentra sobre ningún elemento nada ocurrirá, es decir, no habrá modificación sobre la cinta.

## Entrada

Habrà una línea por cada caso de prueba. La línea de un caso de prueba se compone de una serie de operaciones sobre una máquina de turing de tres cabezas lectoras que inicia con una cinta vacía. Toda operación se compone de al menos dos elementos, primero la cabeza lectora (0 indica la cabeza lectora C0, 1 para C1 y 2 para C2) y luego la operación a realizar (0 indica lectura, 1 para agregar y 2 para eliminar). Si la operación es agregar un elemento entonces tendrá adicionalmente el elemento que se agregará a la cinta. Ni las operaciones, ni los elementos dentro de las operaciones se encuentran separados por ningún carácter. Los elementos a agregar siempre serán letras en mayúsculas del alfabeto inglés (de la A a la Z -26 letras-). Cada caso de prueba tendrá a lo sumo 2147486 caracteres. El archivo de entrada puede pesar cerca de 20MB.

## Salida

Habrà una línea por cada letra leída en cada caso de prueba presente en la entrada. Para cada caso de prueba se imprimirán los elementos leídos por las cabezas lectoras en el orden en que fueron llevadas a cabo estas operaciones.

## Ejemplo

Entrada	Salida
01A1021N11L1000122002 01P21P11I01G 01T1021U11R2021K1011N01I21G001020	A L A N T U R I N G

Usted debe probar su solución con [el caso de prueba grande](#). Usted debe tomar el tiempo que tarda su solución en milisegundos, desde el inicio del programa, antes de cualquier otra instrucción, hasta que todas las demás instrucciones han sido ejecutadas.

La lectura de los datos debe hacerse utilizando archivos de texto (se sugiere `BufferedReader`) y la escritura de la salida también en archivos de texto (se sugiere `BufferedWriter`). Puede utilizar el programa para comparar archivos que se encuentra en el mismo directorio de los archivos de prueba.

El tiempo que tarda su algoritmo debe ser el único valor impreso en consola. El tiempo que tarda para cualquier entrada (la del ejemplo o la grande) no debe superar 1 segundo.

### **FUNCTIONAL REQUIREMENTS**

1. The program should be in the capacity of adding a new element of the linked list by having the position where the element is going to be added. As a result, the element is added to the linked list in the reference passed by parameter.
2. The program should be in the capacity of reading an element of the linked list by having the reference of the element to be read. As a result, the element is read in printed in a text file with the symbol the element contains.
3. The program should be in the capacity of deleting an element of the linked list by having the reference of the element to be deleted. As a result, the element is deleted of the linked list and the reference it's now in another element.

### **NO FUNCTIONAL REQUIREMENTS**

1. The capacity of the program should be measured by taking the time when it starts to read the text file until it finishes.
2. It should be written in a new text file the output of the solution of the method that reads the file.
3. The file the program reads should be in a source folder in the project with the input information.