

Nombre: \_\_\_\_\_ Fecha: \_\_\_\_\_

## Actividad NUMB3RS: Origami de Turing

En "Una hora" Amita y Charlie trabajan tranquilamente en sus propios proyectos cuando Charlie, viendo una máquina 2-D Turing en el laptop de Amita, comenta: "Nunca he visto usar una máquina inductiva de Turing de esa manera". Amita replica que ella está "tratando de averiguar acerca de las máquinas estatales finitas". En esta actividad determinarán la secuencia de pliegue generada por una máquina de Turing.

Antes de que se inventaran las computadoras digitales, el matemático británico Alan Turing propuso un experimento difícil en el que se podían hacer tareas complejas siguiendo una serie de estados condicionales a los que él llamó *máquinas de estados finitos*. Este experimento difícil sentó las bases de la ciencia computacional y muchos de los conceptos que se usan en la programación. Hoy en día, muchas veces pensamos en esas máquinas de estados finitos como líneas de código fuente, en el que cada línea completa una operación determinada. Después de eso, el programa sigue hacia la próxima línea de código. Las máquinas de Turing pueden emplearse para generar códigos. En el experimento de Turing, esas máquinas de estados finitos pueden ser artefactos mecánicos o personas haciendo determinadas tareas.

Para ilustrar el proceso, imagina que entras a la oficina de la FBI en Atlanta y Amita, Charlie y Don están sentados en escritorios diferentes. Tu tarea es reunir una serie de firmas de personas en el orden en que sus nombres aparecen en una hoja de papel. El problema es que cada vez que recibes la firma de alguien, esa persona suma dos nombres nuevos al final de la lista. Una persona no puede, sin embargo, sumar nombres a la lista arbitrariamente, sino siguiendo las siguientes reglas:

- Amita suma a Don y a Charlie
- Charlie suma a Don y a Amita
- Don suma Megan y a Amita

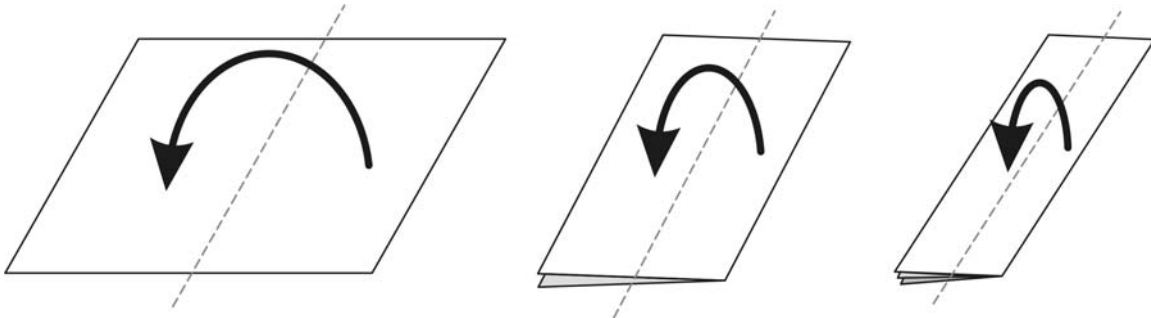
Tu hoja de papel empieza con el nombre de Amita, por lo que primero la visitas en su escritorio. Lo que pasa a continuación se puede representar simbólicamente con las letras A, C, D y M.

Descripción	Simbología
Empiezas con Amita y ella suma a Don y a Charlie a tu lista.	A suma DC para obtener <u>ADC</u>
Luego vas a Don, que suma a Megan y a Amita a tu lista.	<u>ADCMA</u>
Después vas a Charlie, que suma a Don y a Amita a tu lista.	<u>ADCMADA</u>
Después se supone que vas adonde Megan, pero ella no está en la oficina, por lo que paras.	PARAR

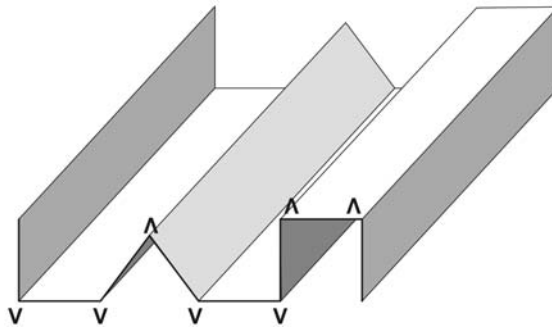
El proceso de generar esta lista es la *máquina de Turing*; las reglas que especifican los nombres que se suman son las *máquinas de estados finitos*; y el escritorio que estás visitando es el "estado" de la máquina de Turing.

Esta es sólo una de las configuraciones posibles. Otras máquinas de Turing pueden tener un número diferente de nombres que se pueden sumar; puede haber nombres que se borran o se puede generar cualquier variante imaginable. Todas las máquinas de Turing tienen estados condicionales (o las "máquinas de estados finitos")

Si doblas repetidamente un papel en su mitad de derecha a izquierda, crearás una serie de dobleces aparentemente aleatorios.



Por ejemplo, si haces los tres dobleces que se muestran arriba y después desdoblas el papel, el resultado será una serie de valles (V) y montañas (Λ), como se muestra a continuación:



La secuencia de dobleces resultante es:

V V Λ V V Λ Λ

Es difícil doblar una papel más de cuatro veces de derecha a izquierda, pero existe una máquina de Turing que generará la secuencia de dobleces usando una serie de máquinas de estados finitos. Esta máquina de Turing, como nuestro ejemplo anterior en que usamos nombres, se fija en un símbolo cada vez y, basada en las máquinas de estados finitos, adiciona dos símbolos al final de la lista.

1. Como nuestra máquina de Turing adiciona dos símbolos al final de la lista, ¿cuáles son las 4 combinaciones posibles que se pueden adicionar usando los símbolos V y Λ?

Como leemos uno de los **dos** símbolos (p. ej., "examina el estado"), entonces adicionamos una de **cuatro** combinaciones, y encontramos un problema. Podemos ver cómo este problema afecta nuestra secuencia en las primeras iteraciones:

Empezando con  $\underline{V}$ , el estado es  $V$ , y el par de combinaciones que se suman son  $V \wedge$ .



Examinando el segundo símbolo  $V \underline{V} \wedge$ , el estado es otra vez  $V$ , pero el par que se suma es  $V V$ .



El problema está en el hecho de que una máquina de estados finitos tiene que ser: **“Cuando el estado es  $V$ , sumar  $V \wedge$  al final de la lista,”** mientras que otra máquina de enunciados finitos es **“Cuando el estado es  $V$ , sumar  $V V$  al final de la lista”**.

Para superar esta contradicción, necesitamos dos variantes de cada símbolo. Una solución es traducir los símbolos a números, de tal manera que los números impares 1 y 3 representen las dos variantes de  $V$ , y los números pares 2 y 4 representen las dos variantes de  $\wedge$ .

Usando estos números para representar las variantes de los símbolos, tenemos las siguientes reglas:

- Cuando el estado es 1, suma 3, 2 al final de la lista y mueve un espacio hacia la derecha.
- Cuando el estado es 2, suma 4, 2 al final de la lista y mueve un espacio hacia la derecha.
- Cuando el estado es 3, suma 3, 1 al final de la lista y mueve un espacio hacia la derecha.
- Cuando el estado es 4, suma 4, 1 al final de la lista y mueve un espacio hacia la derecha.

Con estas máquinas de estados finitos en su lugar, la secuencia de dobleces del papel se genera como sigue:

Empezar con 1.

El primer estado es 1, así que suma 3, 2 al final de la lista: **1, 3, 2**

El próximo estado es 3, así que suma 3, 1 al final de la lista: **1, **3**, 2, 3, 1**

2. Continúa hasta que hayas generado todas las 15 cifras para la cuarta secuencia de dobleces.

--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

El paso final es traducir esta secuencia de Turing hacia atrás, a la secuencia de dobleces del papel con símbolos de  $V$  y  $\wedge$ .



*El objeto de esta actividad es dar a los estudiantes un vistazo breve y sencillo de un tema matemático muy extenso. TI y NCTM lo invitan a usted y a sus estudiantes a aprender más sobre este tema con las extensiones que se ofrecen abajo y con su propia investigación independiente.*

### Extensiones

1. Completa la tabla de la derecha y compara el número de dobleces al número de dobles.
2. Halla las diferencias entre los números de dobleces.
3. Escribe esas diferencias como potencias de la misma base.

Doblez	Número de dobleces
1	1
2	3
3	7
4	
5	
6	
7	

4. Expresa esas diferencias como una sumatoria.

$$\sum_{i=1}^n \text{_____}$$

Abajo aparece una sección de un programa para una calculadora gráfica que genera secuencias de Turing y secuencia de dobleces para cualquier número de dobles. Identifica la sumatoria que se describió y también las máquinas finitas que se encuentran en la actividad.

```

:ClrHome
:" 1" →Str1
:0 →C
:Disp " ENTER THE NUMBER"
:I nput " OF FOLDS ", N
:sum(seq(2^(X-1), X, 1, N) →F
:While length(Str1) < F
:C+1 →C
:I f sub(Str1, C, 1) = " 1"
:Str1 + " 32" →Str1
:I f sub(Str1, C, 1) = " 2"
:Str1 + " 42" →Str1
:I f sub(Str1, C, 1) = " 3"
:Str1 + " 31" →Str1
:I f sub(Str1, C, 1) = " 4"
:Str1 + " 41" →Str1
:End
:ClrHome
:Disp " TURING SEQUENCE"
:Disp Str1
    
```