Informe: Torres de Hanoi

Isaac Aimán Salas

Introducción

En este informe se analiza el juego matemático de *Las Torres de Hanoi*. Para ello, en los diferentes apartados del mismo se llevará a cabo una descripción del problema en sí, se aportará el pseudocódigo del algoritmo recursivo que resuelve el rompecabezas y se analizará la complejidad del algoritmo que lo implementa.

Descripción del problema

El juego *Las Torres de Hanoi* es un rompecabezas que consta de tres varillas y un número determinado de discos. El objetivo es colocar todos los discos, que en un principio se encuentran ordenados por tamaño en una varilla, en alguna de las otras dos varillas que inicialmente se encuentran vacías. A continuación, se detallan las reglas del juego:

- Solo se puede mover un disco a la vez.
- Solo se puede colocar un disco encima de otro de mayor tamaño.

Este juego presenta una variante denominada *Las Torres de Hanoi Cíclicas*, que analizaremos más adelante.

Pseudocódigo

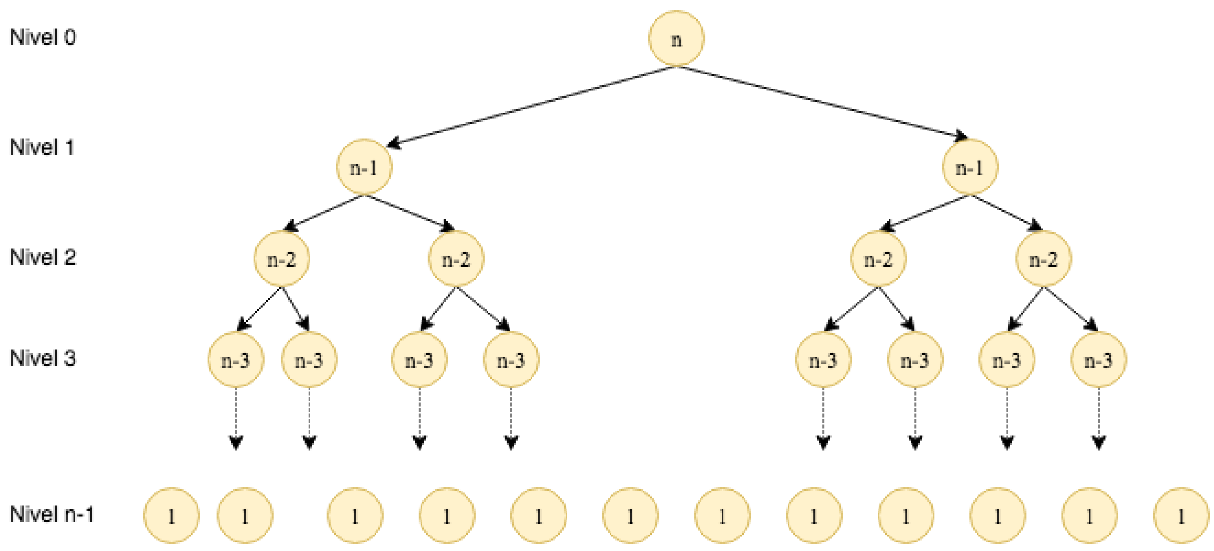
Para resolver este problema, haremos uso de un algoritmo recursivo que sigue el paradigma **divide y vencerás**. En cada llamada recursiva el problema se divide en dos subproblemas de tamaño n-1, siendo n el número total de discos con los que se pretende resolver el juego.

A continuación, se muestra el pseudocódigo de la función recursiva:

```
Entrada: el número de discos y tres varillas (varilla de origen, varilla auxiliar y varilla de destino)
Si el número de discos == 1 entonces
     mover un disco de la varilla de origen a la varilla de destino
fin
Si no
     hanoi(número de discos - 1, varilla origen, varilla destino, varilla auxiliar)
     mover un disco de la varilla de origen a la varilla de destino
     hanoi(número de discos - 1, varilla auxiliar, varilla origen, varilla destino)
fin
```

Análisis de la complejidad

El árbol de recursividad para el algoritmo de *Las Torres de Hanoi* expuesto anteriormente es el siguiente:



Como podemos apreciar en el árbol, el **número de niveles** es n-1, pues la función recursiva deja de llamarse a sí misma al alcanzar un subproblema de tamaño 1 (tal y como puede apreciarse en el pseudocódigo proporcionado).

El **número total de subproblemas por nivel** es 2^i, pues en cada llamada a la función, el problema se divide en dos subproblemas de tamaño n-i.

La **complejidad** del algoritmo viene determinada por el siguiente sumatorio:

$$T(n) = \sum_{i=0}^{n-1} 2^i = 2^n - 1$$

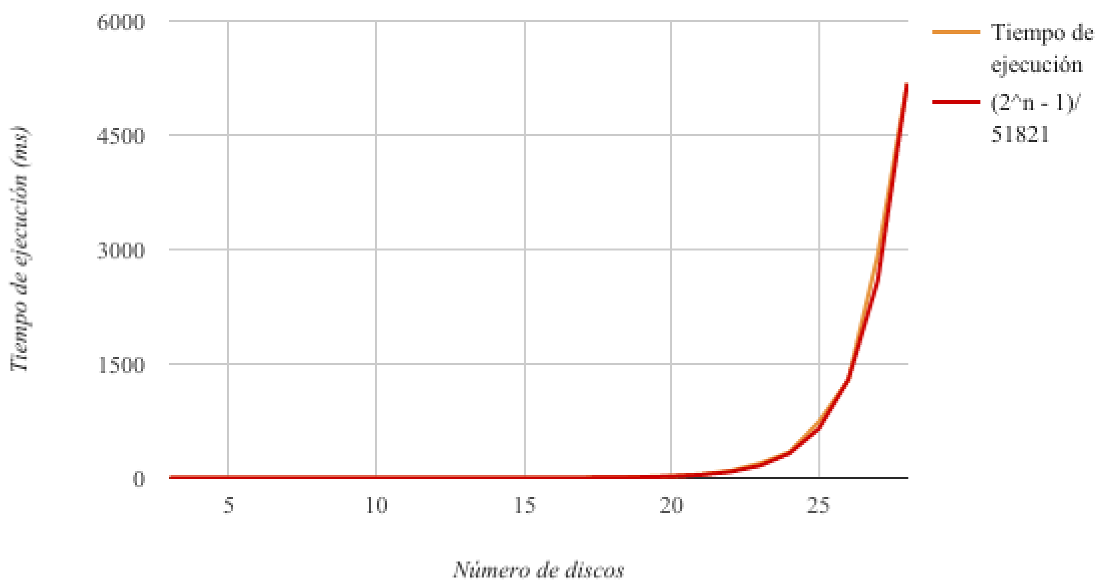
Para este algoritmo, puesto que no existe la posibilidad de un mejor o un peor caso, la complejidad es siempre 2^n - 1.

A continuación, se detallan los **tiempos de ejecución** del algoritmo para diferentes números de discos:

Número de discos	Tiempo en ms	Número de discos	Tiempo en ms
3	0,049712	16	1,218273
4	0,071657	17	2,884377
5	0,061835	18	5,862144
6	0,150515	19	13,633946
7	0,118316	20	24,446003

8	0,332865	21	47,187005
9	0,103857	22	92,88735
10	0,138512	23	191,774278
11	0,236104	24	339,040797
12	0,351723	25	738,908692
13	0,773921	26	1290,794831
14	1,695146	27	2927,745207
15	2,125423	28	5180,119895

Si representamos gráficamente los datos expuestos anteriormente y los comparamos con los obtenidos teóricamente, obtenemos la siguiente gráfica:



Pseudocódigo: Hanoi Cíclico

Las Torres de Hanoi Cíclicas, es una variante de *Las Torres de Hanoi*. En esta nueva modalidad de juego, los discos tienen que moverse en el sentido de las agujas del reloj. Por tanto, para mover un disco de la varilla 1 a la 3 antes deberá pasar por la varilla 2. Una vez alcanzada la última varilla, el siguiente movimiento es a la varilla 1.

A continuación, se muestra el pseudocódigo de esta variante de *Las Torres de Hanoi*:

Para implementar esta nueva modalidad del juego son necesarias dos funciones. La primera de ellas es HanoiSentidoHorario:

```
Entrada: el número de discos y tres varillas (varilla a, varilla b y varilla c)
Si el número de discos > 0 entonces
     HanoiSentidoAntiHorario(numero de discos - 1, a, c, b)
     Mover disco a de la varilla ‘a’ a la siguiente varilla en sentido horario.
     HanoiSentidoAntiHorario(numeroDiscos - 1, c, b, a)
fin
```

La segunda función necesaria será HanoiSentidoAntiHorario:

```
Entrada: el número de discos y tres varillas (varilla a, varilla b y varilla c)
Si el número de discos > 0 entonces
     HanoiSentidoAntiHorario(numeroDiscos - 1, a, b, c);
     Mover disco a de la varilla ‘a’ a la siguiente varilla en sentido horario.
     HanoiSentidoHorario(numeroDiscos - 1, b, a, c);
     Mover disco a de la varilla ‘c’ a la siguiente varilla en sentido horario.
     HanoiSentidoAntiHorario(numeroDiscos - 1, a, b, c);
fin
```

Conclusión

Podemos observar, por los datos obtenidos experimentalmente y por el análisis teórico llevado a cabo, que el algoritmo para resolver el juego de las *Torres de Hanoi* es su modalidad simple, posee unos tiempos de ejecución relativamente bajos para pocos discos, pero este tiempo aumenta exponencialmente con el número de discos. Por este motivo, la eficiencia del algoritmo para un número considerable de discos es muy baja (tal y como se puede apreciar en la gráfica expuesta en el apartado del análisis de complejidad).

Referencias bibliográficas

Cormen, T., Leiserson, C., Rivest, R., & Stein, C. (2014). *Introduction to algorithms*. Cambridge, Massachusetts: The MIT Press.

Tower of Hanoi. (2017). En *wikipedia.org*. Revisado el 5 de marzo de 2017, desde https://en.wikipedia.org/wiki/Tower_of_Hanoi

Atkinson, M. (1981). The cyclic towers of Hanoi. *Information Processing Letters*, *13*(3), 118-119. http://dx.doi.org/10.1016/0020-0190(81)90123-x