Recursividad

parte II

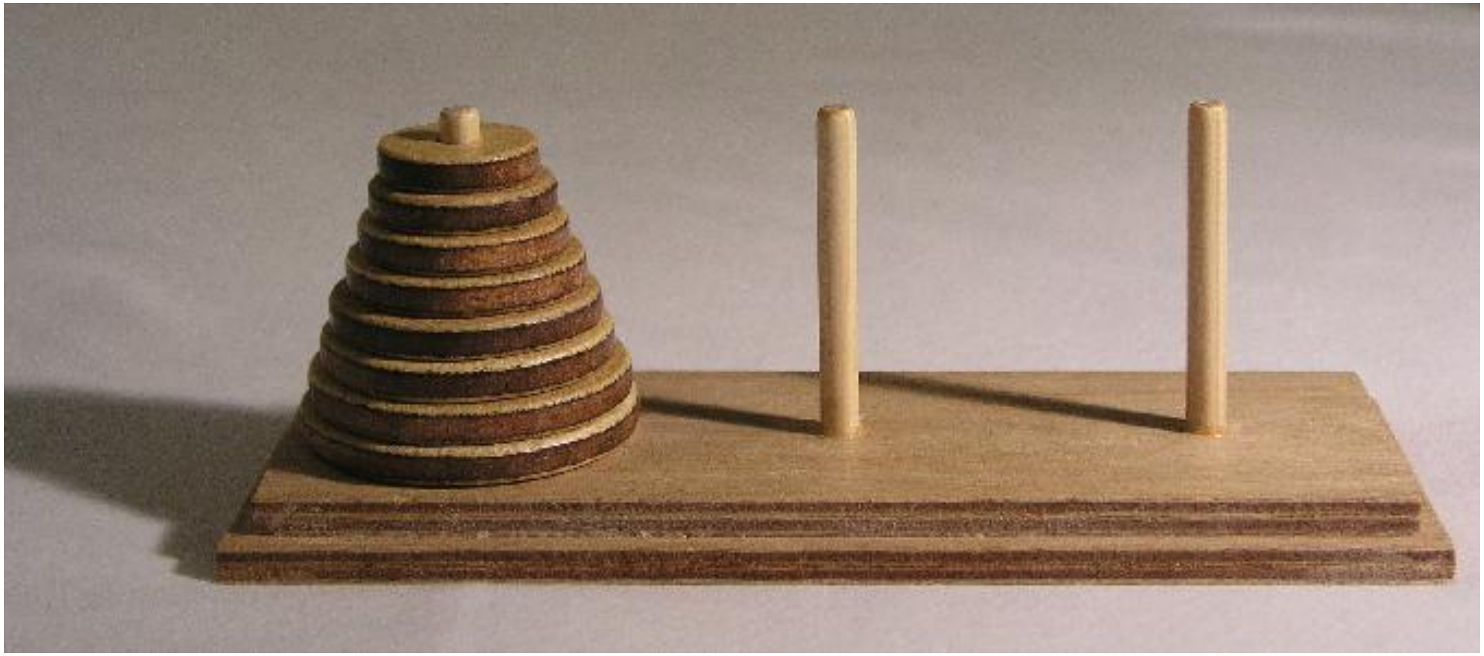
Departamento de Computación

Algoritmos y Programación I

Joel Rivas

Torres de Hanoi

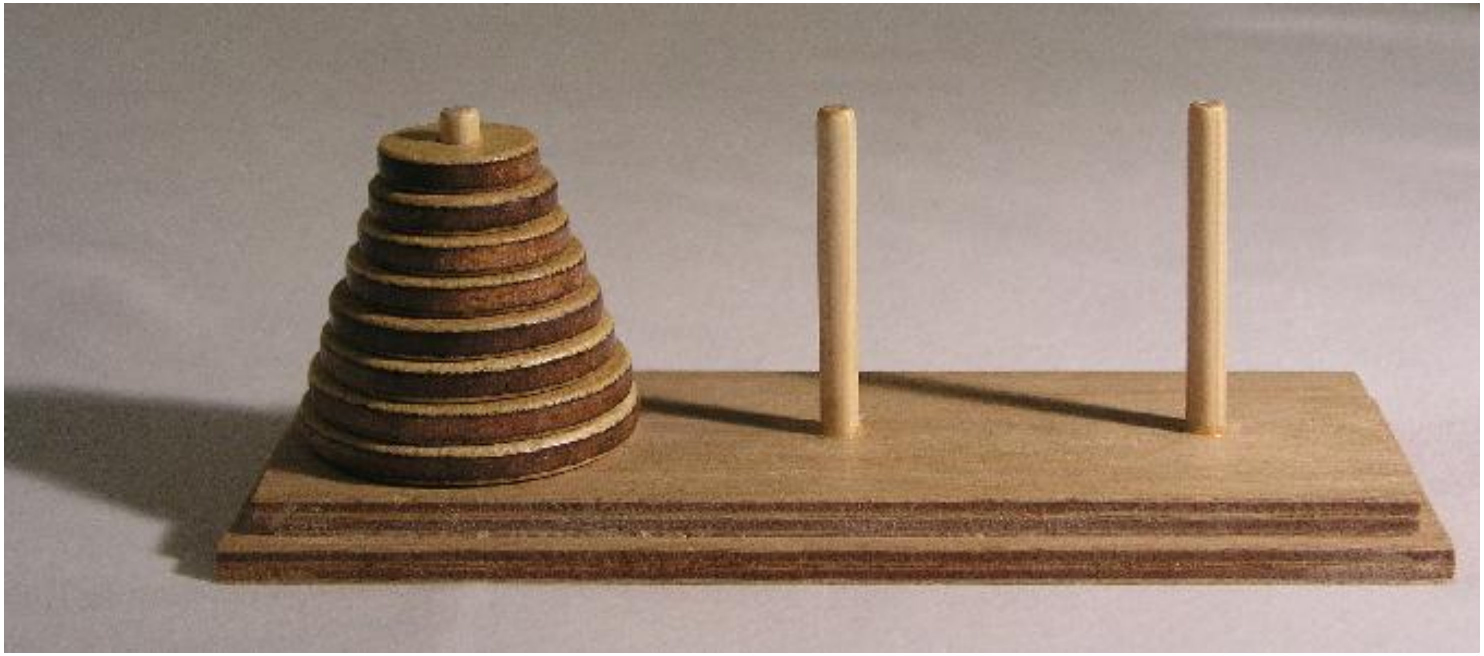
- Las Torres de Hanoi es un rompecabezas o juego matemático dado a conocer en 1883 por el matemático francés Éduard Lucas.
- El juego consiste en una base con tres postes (A, B, C) y un grupo de discos de diferentes diámetros que se apilan insertados en el poste A, unos sobre otros en orden decreciente del diámetro, de abajo hacia arriba (el número de discos puede variar).



Torres de Hanoi: El Problema

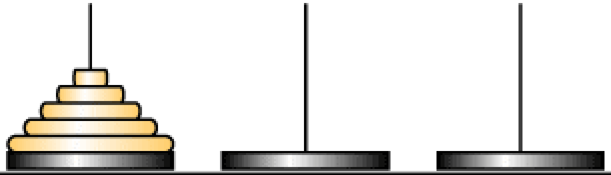
- El Juego consiste en pasar todos los discos del poste A al poste C, usando el otro poste como auxiliar y respetando las siguientes reglas:

  1.- Sólo un disco se puede mover a la vez.
  2.- Un disco nunca puede descansar sobre otro disco que tenga diámetro menor.
  3.- Un disco siempre debe estar en uno de los postes



Torres de Hanoi: La Leyendas

"En el gran templo de Benarés (Uttar Pradesh, India), bajo la cúpula que marca el centro del universo, hay un plato de bronce con tres agujas de diamante, tan finas como el cuerpo de una abeja. En una de esas agujas, la de la creación, Dios colocó 64 discos de oro puro, el mayor de ellos descansaba en el plato de bronce y los demás haciéndose cada vez menores hasta llegar al más pequeño encima de todos. Día y noche sin parar, los monjes mueven los discos de una aguja de diamante a otra, de acuerdo con las sagrada leyes, las cuales requieren que cada monje tenga que mover un solo disco cada vez de una de la agujas a otra aguja de forma que no lo deposite sobre un disco menor. Cuando los 64 discos sean trasladados de la aguja en la que Dios los depositó en el momento de la creación hacia la aguja final; la torre, el templo y los monjes se convertirán en polvo, y con un trueno el mundo desaparecerá".



Torres de Hanoi: Abordando el problema

El problema de las Torres de Hanoi generalmente se trabaja con menos discos, normalmente 8 discos.

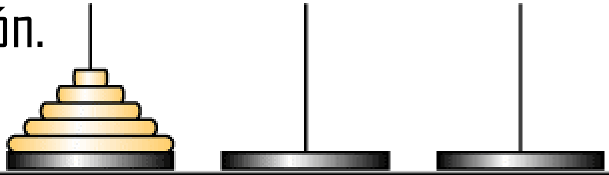
El problema consiste por tanto en trasladar la torre a otra varilla, moviendo un disco cada vez, de manera que en ningún momento un disco descanse sobre otro de menor tamaño.

Inicialmente sólo es posible mover el disco de menor tamaño. El segundo movimiento también está forzado. A partir del tercer movimiento, la elección ya no es única.

Se ha demostrado que para mover n discos son necesarios mínimo

$2^n – 1$ movimientos.

El número de discos determina la complejidad de la solución.



Torres de Hanoi: Complejidad

Así, el número mínimo de "movimientos" para conseguir transferir todos los discos es $\mathbf{2^n - 1}$, siendo n el número de discos.

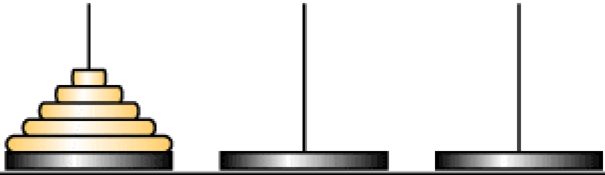
Por lo tanto:

Para 3 discos , son necesarios 7 movimientos

Para 7 discos, son necesarios 127 movimientos

Para 15 discos, son necesarios 32.767 movimientos

Para 64 discos, como dice la leyenda, son necesarios 18.446.744.073.709.551.615 movimientos.



Torres de Hanoi: Solución iterativa

Con número impar de discos:

El primer movimiento debe ser colocar el disco más pequeño en la pila destino.

En cada paso impar se le mueve a la siguiente pila a su izquierda (o a la pila destino, si está en la pila origen).

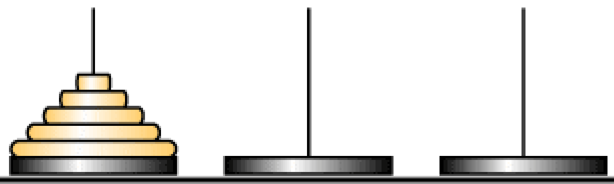
La secuencia será DESTINO, AUXILIAR, ORIGEN, DESTINO, AUXILIAR, ORIGEN, etc.

Con número par de discos:

El primer movimiento debe ser colocar el disco más pequeño en la pila auxiliar.

En cada paso impar se le mueve a la siguiente pila a su derecha (o a la pila origen, si está en la pila destino).

La secuencia será AUXILIAR, DESTINO, ORIGEN, AUXILIAR, DESTINO, ORIGEN, etc.



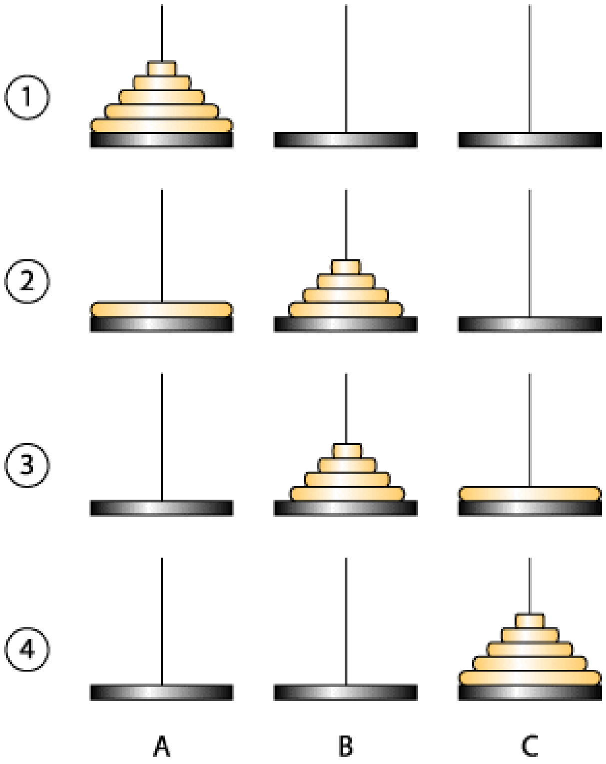
Torres de Hanoi: Solución Recursiva

Algoritmo para mover n discos de A hacia C, usando B como auxiliar:

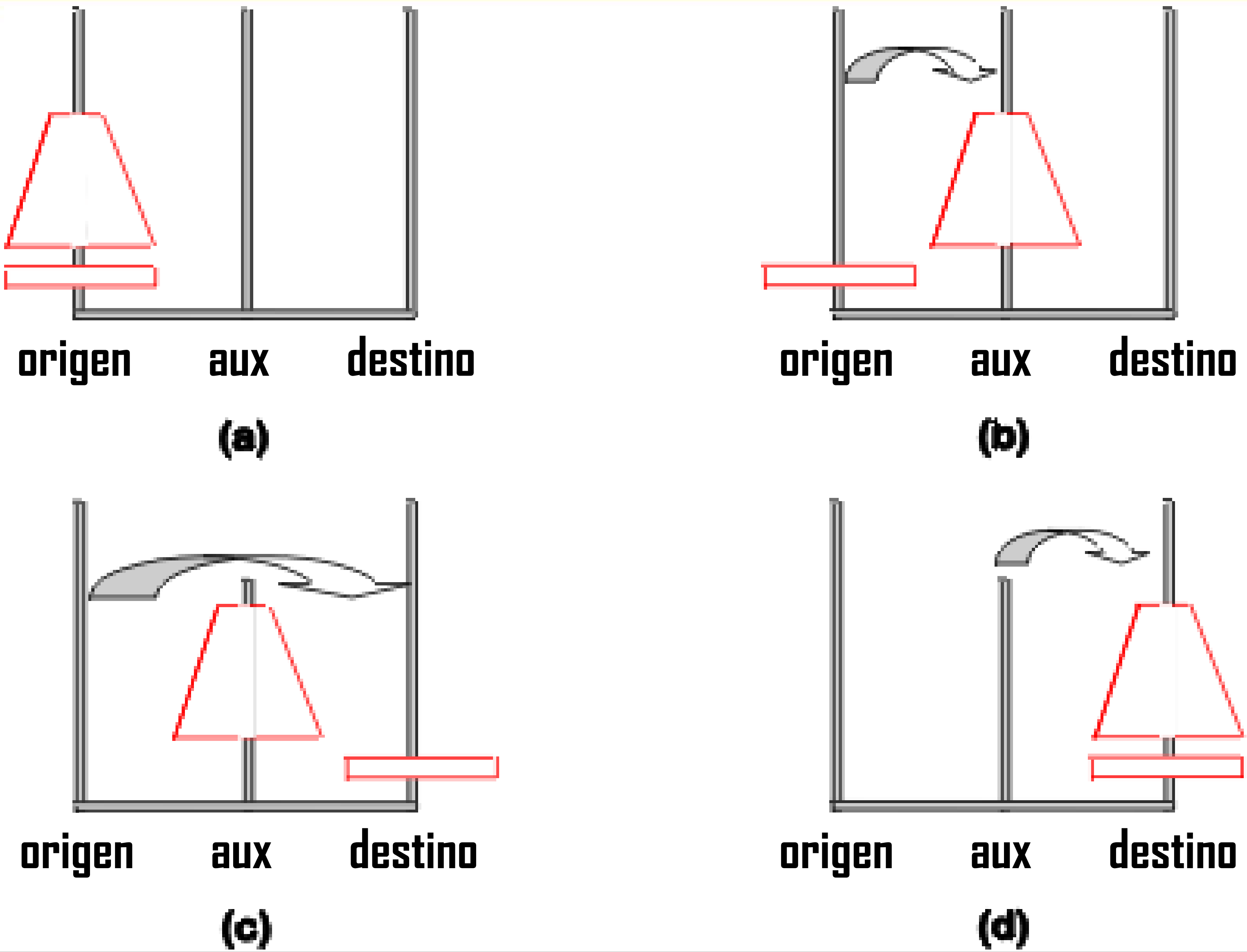
Si n=1, mover el único disco de A hacia C y parar.

Si n>1 entonces

1. Mover los n–1 discos superiores de A hacia B, usando C como auxiliar.
2. Mover el disco restante de A hacia C.
3. Mover los n–1 discos de B hacia C, usando A como auxiliar.



Torres de Hanoi: Solución Recursiva



Torres de Hanoi: Algoritmo

```
Algoritmo Torres_de_Hanoi

proc hanoi(entero n, caracter desde, caracter hasta,
           caracter aux, ref entero nmov)
inicio
 var  // variables locales
  entero    num_discos, num_movim

  escribir("Programa para Resolver el problema de las
           Torres de Hanoi")
  // lectura de num_discos y validación num_discos >= 1
  leeValida(num_discos, "" numero de discos", 1)

  // Pasos para transferir los discos
  num_movim ← 0
  hanoi(num_discos, 'A', 'C', 'B', num_movim)

  escribir("Número total de movimientos", num_movim)
fin   // fin de principal
```

```
proc hanoi(entero n, caracter desde, caracter hasta,
           caracter aux, ref entero nmov)
inicio
  si (n = 1) entonces
     nmov ← nmov + 1
     escribir(nmov, ".- mover disco de ", desde," hacia ",
hasta)
  sino
     hanoi(n-1 , desde, aux, hasta, nmov)
     nmov ← nmov + 1
     escribir(nmov, ".- mover disco de ",desde," hacia ",
hasta)
     hanoi(n-1 , aux, hasta, desde, nmov)
fproc   // fin hanoi
```