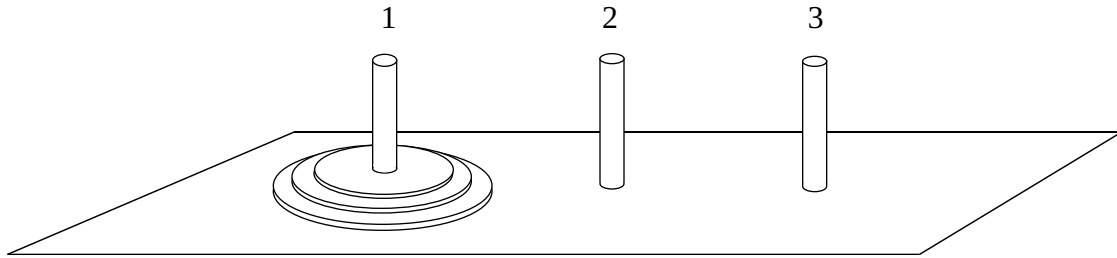


# Torres de Hanói

Arquivo fonte: *hanoi.c*, *hanoi.cc*, *hanoi.cpp* ou *hanoi.pas*

O quebra-cabeças Torres de Hanoi é muito antigo e conhecido, sendo constituído de um conjunto de  $N$  discos de tamanhos diferentes e três pinos verticais, nos quais os discos podem ser encaixados.



Cada pino pode conter uma pilha com qualquer número de discos, desde que cada disco não seja colocado acima de outro disco de menor tamanho. A configuração inicial consiste de todos os discos no pino 1. O objetivo do quebra-cabeças é mover todos os discos para um dos outros pinos, sempre obedecendo à restrição de não colocar um disco sobre outro menor.

Um algoritmo para resolver este problema é o seguinte.

```
procedimento Hanoi( $N$ , Orig, Dest, Temp)
  se  $N = 1$  então
    mover o menor disco do pino Orig para o pino Dest;
  senão
    Hanoi( $N-1$ , Orig, Temp, Dest);
    mover o  $N$ -ésimo menor disco do pino Orig para o pino Dest;
    Hanoi( $N-1$ , Temp, Dest, Orig);
  fim-se
fim
```

## 1. Tarefa

Sua tarefa é escrever um programa que determine quantos movimentos de trocar um disco de um pino para outro serão executados pelo algoritmo acima para resolver o quebra-cabeça.

## 2. Entrada

A entrada possui vários conjuntos de teste. Cada conjunto de teste é composto por uma única linha, que contém um único número inteiro  $N$  ( $0 \leq N \leq 30$ ), indicando o número de discos. O final da entrada é indicado por  $N = 0$ .

### Exemplo de Entrada

```
1
2
0
```

### 3. Saída

Para cada conjunto de teste, o seu programa deve escrever três linhas na saída. A primeira linha deve conter um identificador do conjunto de teste, no formato “Teste  $n$ ”, onde  $n$  é numerado sequencialmente a partir de 1. A segunda linha deve conter o número de movimentos que são executados pelo algoritmo dado para resolver o problema das Torres de Hanói com  $N$  discos. A terceira linha deve ser deixada em branco. A grafia mostrada no Exemplo de Saída, abaixo, deve ser seguida rigorosamente.

#### Exemplo de Saída

Teste 1

1

Teste 2

3

(esta saída corresponde ao exemplo de entrada acima)

### 4. Restrições

$0 \leq N \leq 30$  ( $N = 0$  apenas para indicar o final da entrada)