

PROBLEMA:

Transportar n discos empilhados na haste 1 (início) para a haste 3 (destino), usando eventualmente uma haste auxiliar (2) e garantindo que:

- um disco seja transportado de cada vez;
- um disco maior não fique em cima de um disco menor.

ESTRATÉGIA:

- n discos
- início = 1
- destino = 3
- auxiliar = 2

1. Trivial para $n=1$; senão,
2. Mover $(n-1)$ discos de início para auxiliar;
3. Mover o último disco de início para destino;
4. Mover $(n-1)$ discos de auxiliar para destino.

Duas chamadas recursivas reduzem o problema ao caso trivial, memorizando os passos seguintes.

Requer $2^n - 1$ movimentos (cresce exponencialmente com o número de discos. Possui solução recursiva e iterativa.

PROGRAMA C:

```
#include <stdio.h>

void move (int ini, int aux, int des, int n) {
    if (n==1) printf ("Mover de %d para %d\n",ini,des);
    else {
        // printf ("---> move %d de %d para %d usando
%d\n",n-1,ini,aux,des);
        move (ini,des,aux,n-1);
        printf ("Mover de %d para %d\n",ini,des);
        // printf ("---> move %d de %d para %d usando
%d\n",n-1,aux,des,ini);
        move (aux,ini,des,n-1);
    }
}

int main () {
    int n;
    printf ("Digite o número de discos:");
    scanf ("%d",&n);
    move (1,2,3,n);
}
```