

Problema O

Hanói

Nome base: hanoi

Tempo limite: 1s

Em programação lógica, a Torre de Hanói é um dos problemas clássicos para estudar backtracking.

Para relembrar o conceito, neste problema, você precisa encontrar a quantidade de movimentos para trocar um disco de uma torre para outra, dado o algoritmo abaixo.

```
hanoi(Qtd, Origem, Destino, Aux) {  
  se Qtd = 1 {  
    levar disco da torre Origem para torre Destino  
  }  
  senão {  
    hanoi(Qtd-1, Origem, Aux, Destino)  
    levar o Qtd-ésimo disco da torre Origem para torre Destino  
    hanoi(Qtd-1, Aux, Destino, Origem)  
  }  
}
```

ENTRADA

A entrada contém um inteiro X ($0 \leq X \leq 25$) que representa a quantidade de discos.

SAÍDA

A saída é um número inteiro com a quantidade de movimentos realizada pelo algoritmo fornecido no enunciado deste problema, para levar o disco da torre de origem para a torre de destino.

| Exemplo de Entrada | Exemplo de Saída |
|--------------------|------------------|
| 2 | 3 |

| Exemplo de Entrada | Exemplo de Saída |
|--------------------|------------------|
| 4 | 15 |