

# Estrutura de Dados II

## Recursividade

(Aula02 – Torre de Hanói)

Prof. Rafael Nunes

# Torre de Hanói

## Parte 2

# Recursividade e a memória

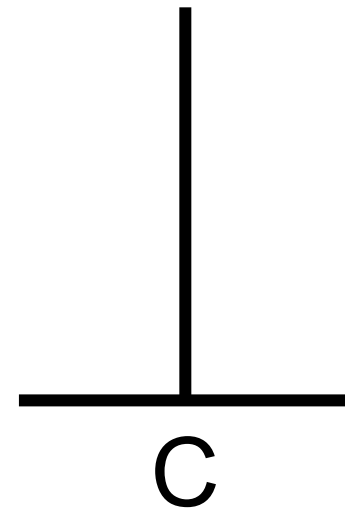
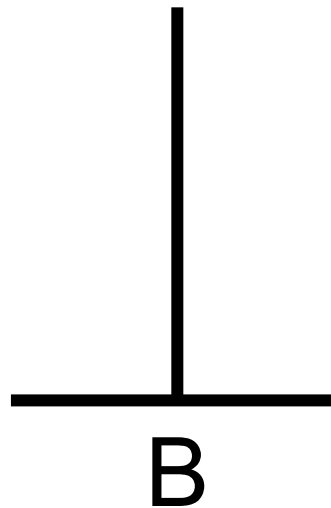
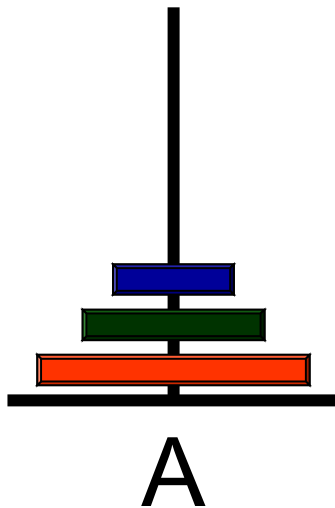
# Torre de Hanói

5		
4		
3		
2		
1		
0	Hanoi_cham(3, A, B, C)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



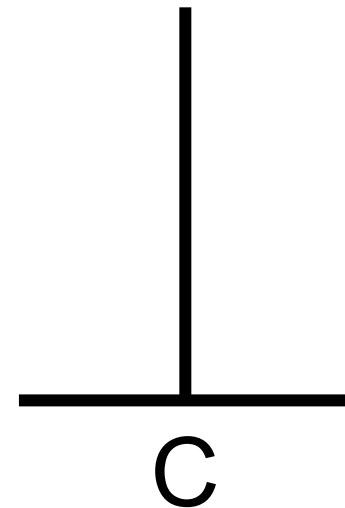
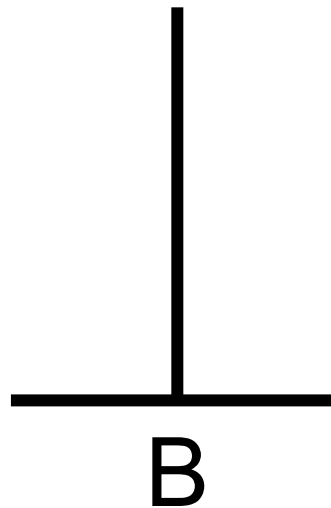
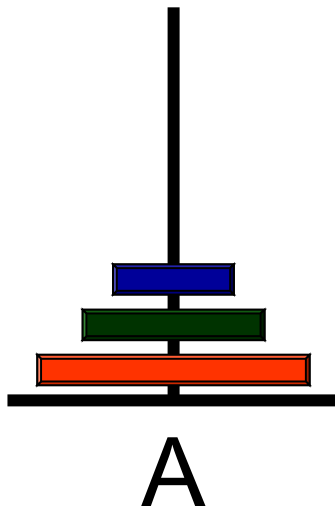
# Torre de Hanói

5		
4		
3		
2		
1	Hanoi_cham(3, A, B, C)	
0	Hanoi_1(2, A, C, B)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



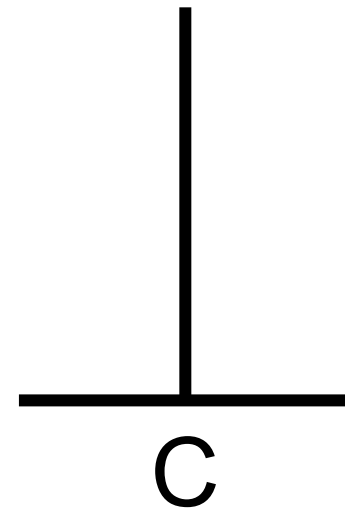
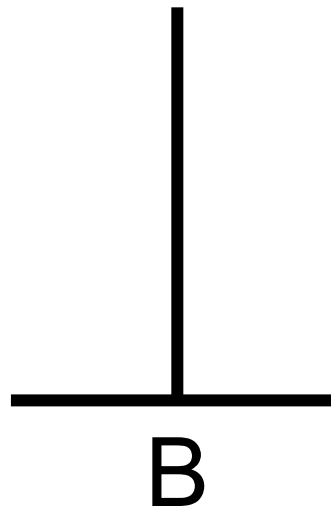
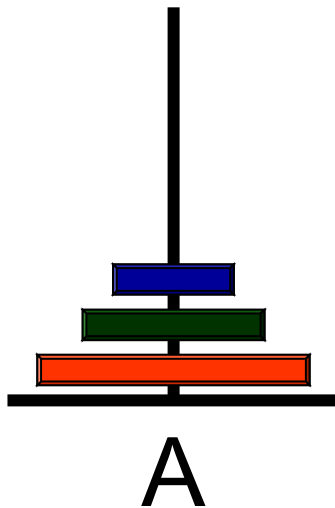
# Torre de Hanói

5		
4		
3		
2	Hanoi_cham(3, A, B, C)	
1	Hanoi_1(2, A, C, B)	
0	Hanoi_1(1, A, B, C)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



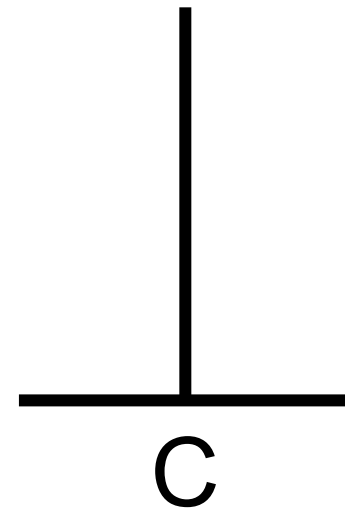
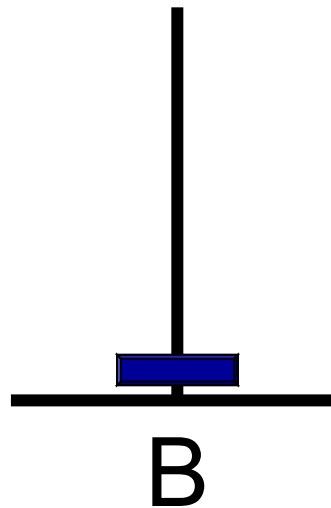
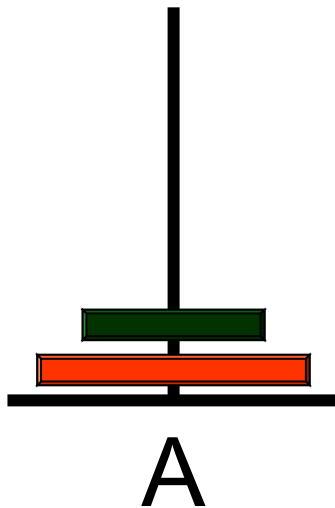
# Torre de Hanói

5		
4		
3		
2	Hanoi_cham(3, A, B, C)	
1	Hanoi_1(2, A, C, B)	
0	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B

**1 = Azul**

**2 = Verde**

**3 = Laranja**



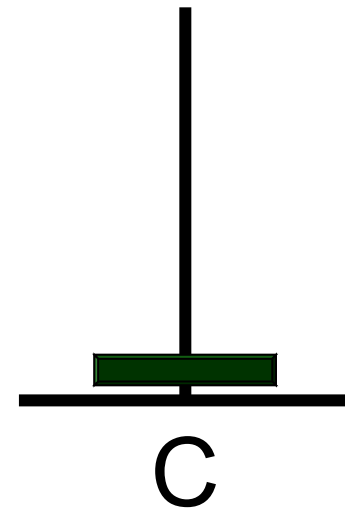
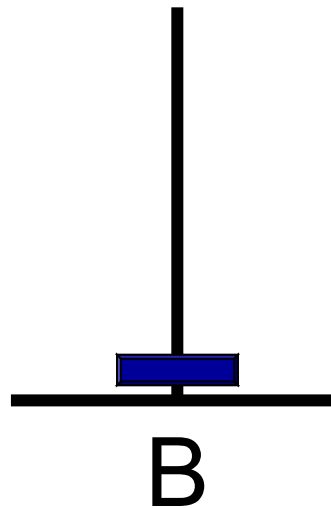
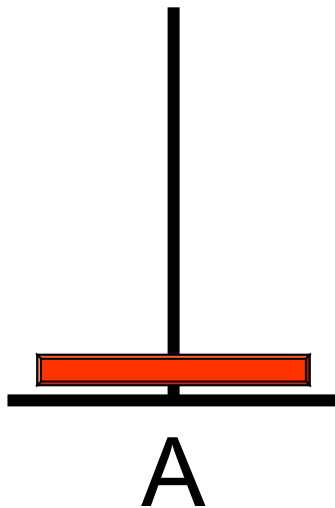
# Torre de Hanói

5		
4		
3		
2	Hanoi_cham(3, A, B, C)	
1	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
0	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B

**1 = Azul**

**2 = Verde**

**3 = Laranja**





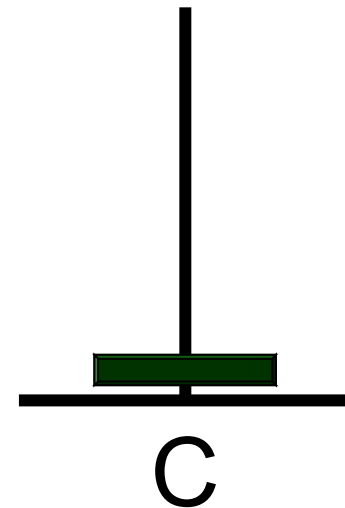
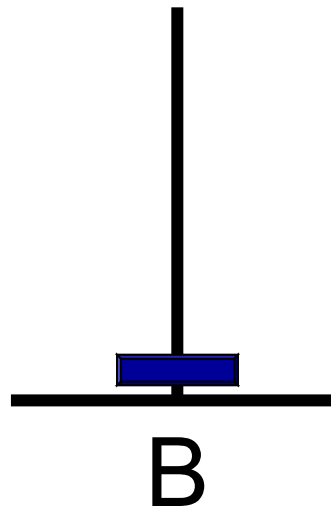
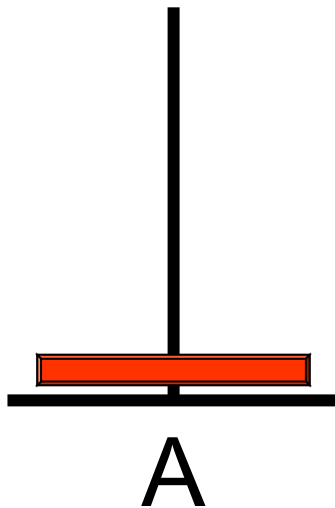
# Torre de Hanói

5		
4		
3	Hanoi_cham(3, A, B, C)	
2	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
1	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
0	Hanoi_cham2(2, A, C, B)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



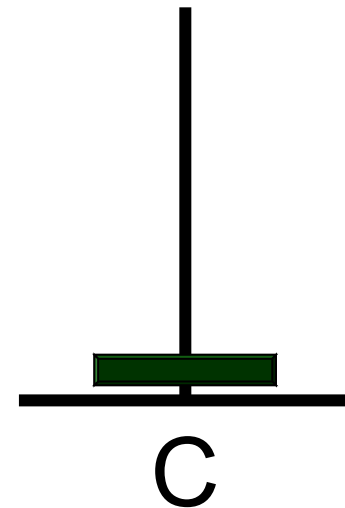
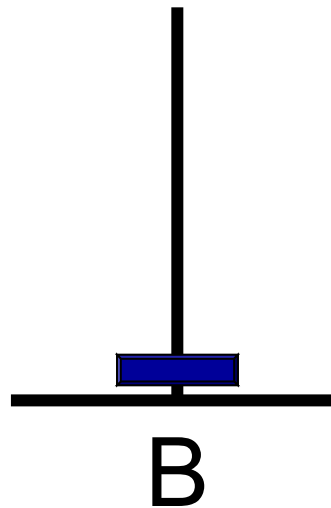
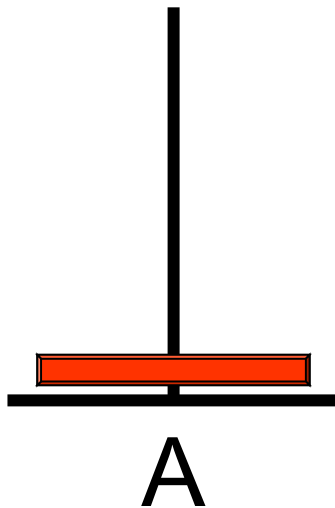
# Torre de Hanói

5		
4	Hanoi_cham(3, A, B, C)	
3	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
2	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
1	Hanoi_cham2(2, A, C, B)	
0	Hanoi_2(1, B, C, A)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



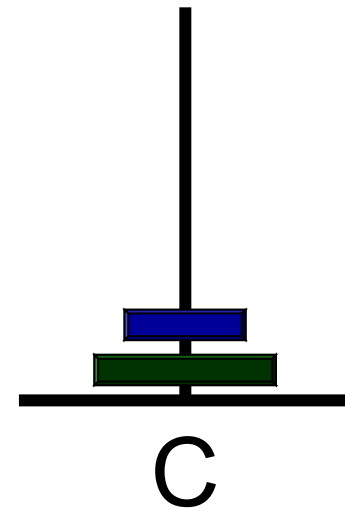
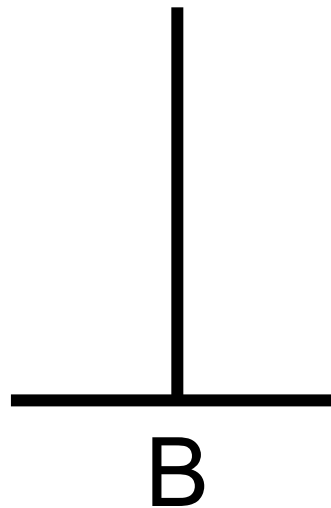
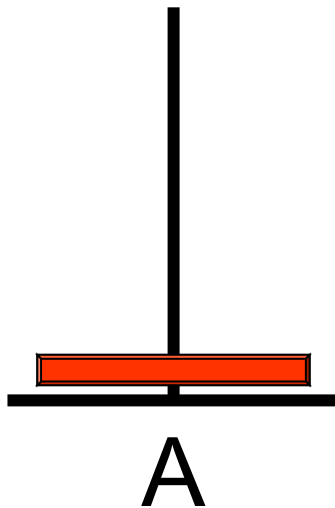
# Torre de Hanói

5		
4	Hanoi_cham(3, A, B, C)	
3	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
2	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
1	Hanoi_cham2(2, A, C, B)	
0	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C

**1 = Azul**

**2 = Verde**

**3 = Laranja**



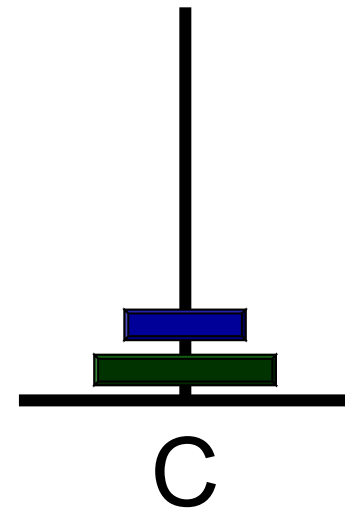
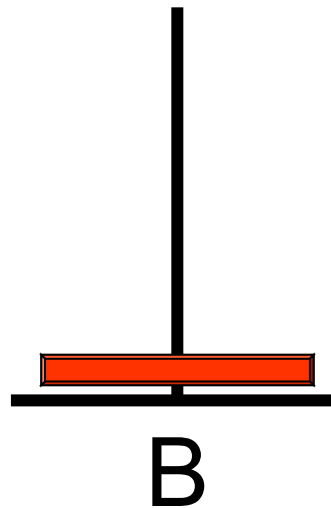
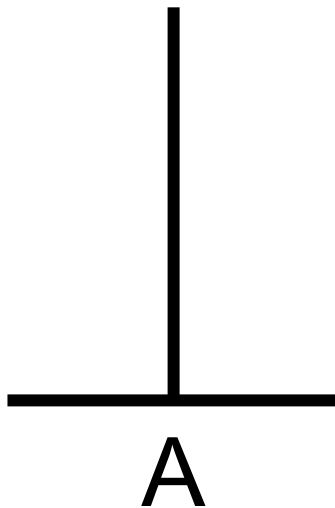
# Torre de Hanói

5		
4	Hanoi_cham(3, A, B, C)	Mover disco 3 de A para B
3	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
2	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
1	Hanoi_cham2(2, A, C, B)	
0	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C

**1 = Azul**

**2 = Verde**

**3 = Laranja**



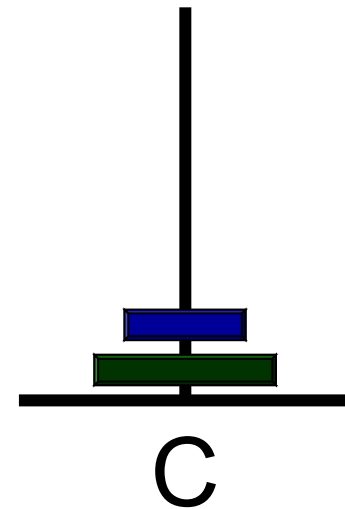
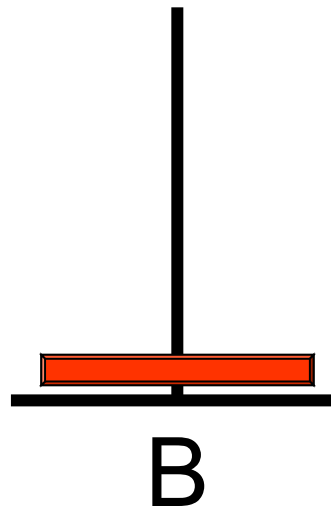
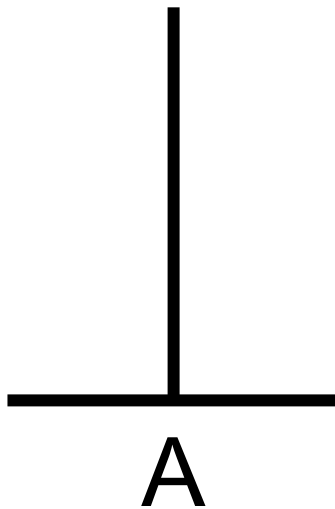
# Torre de Hanói

5	Hanoi_cham(3, A, B, C)	Mover disco 3 de A para B
4	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
3	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
2	Hanoi_cham2(2, A, C, B)	
1	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
0	Hanoi_cham2(3, A, B, C)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



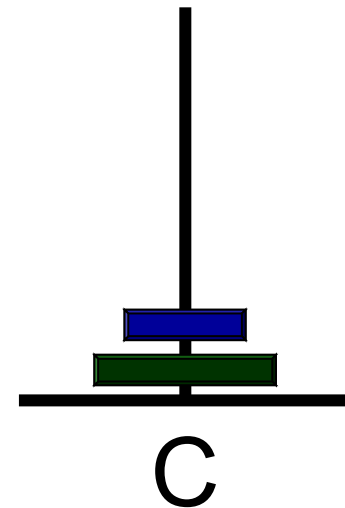
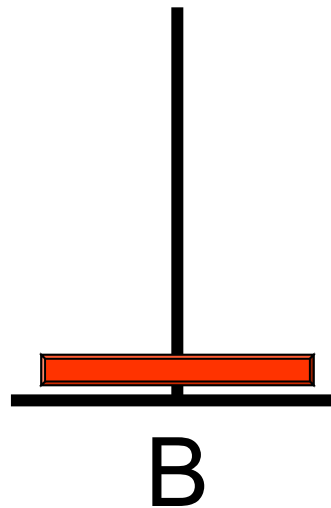
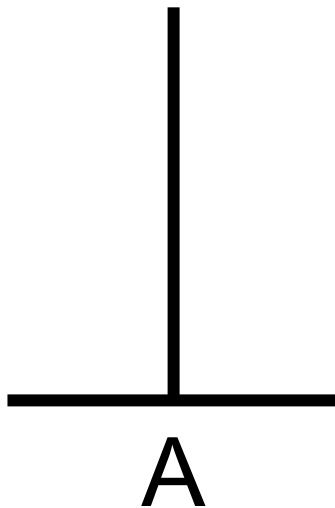
# Torre de Hanói

5	Hanoi_1(2, A, C, B)	Mover disco 2 de A para C
4	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
3	Hanoi_cham2(2, A, C, B)	
2	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
1	Hanoi_cham2(3, A, B, C)	
0	Hanoi_2(2, C, B, A)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



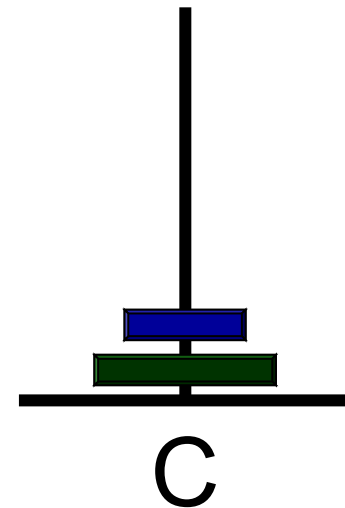
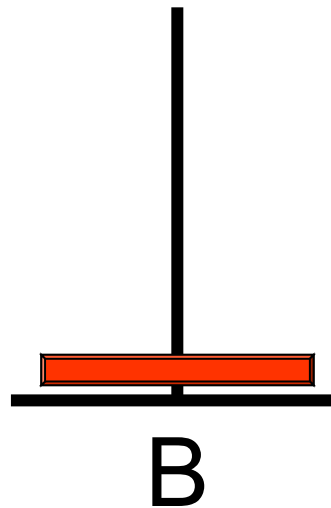
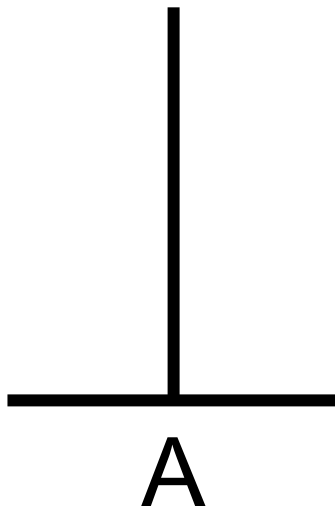
# Torre de Hanói

5	Hanoi_1(1, A, B, C)	Mover disco 1 de A para B
4	Hanoi_cham2(2, A, C, B)	
3	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
2	Hanoi_cham2(3, A, B, C)	
1	Hanoi_2(2, C, B, A)	
0	Hanoi_cham(2, C, B, A)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



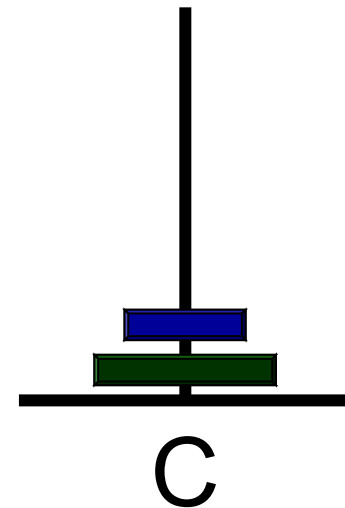
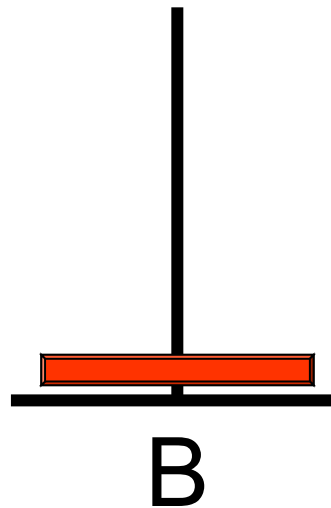
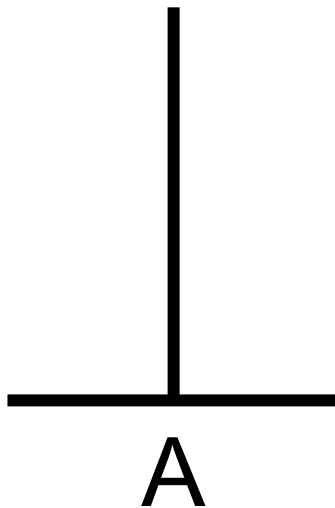
# Torre de Hanói

5	Hanoi_cham2(2, A, C, B)	
4	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
3	Hanoi_cham2(3, A, B, C)	
2	Hanoi_2(2, C, B, A)	
1	Hanoi_cham(2, C, B, A)	
0	Hanoi_1(1, C, A, B)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**





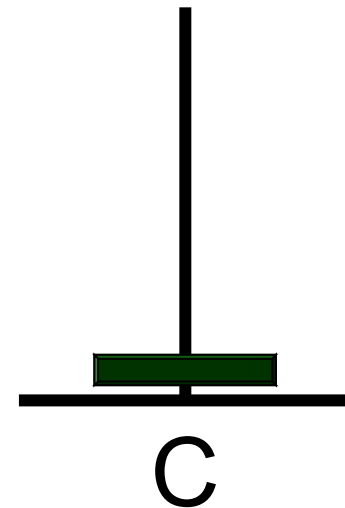
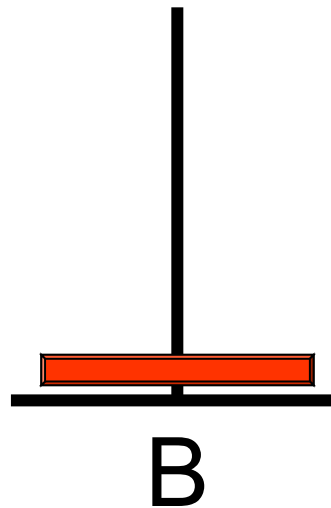
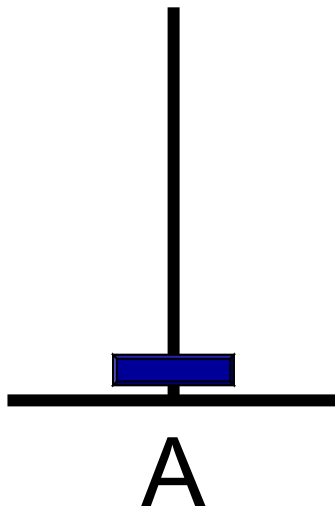
# Torre de Hanói

5	Hanoi_cham2(2, A, C, B)	
4	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
3	Hanoi_cham2(3, A, B, C)	
2	Hanoi_2(2, C, B, A)	
1	Hanoi_cham(2, C, B, A)	
0	Hanoi_1(1, C, A, B)	Mover disco 1 de C para A

**1 = Azul**

**2 = Verde**

**3 = Laranja**



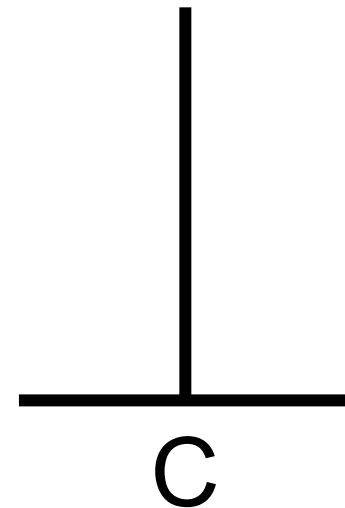
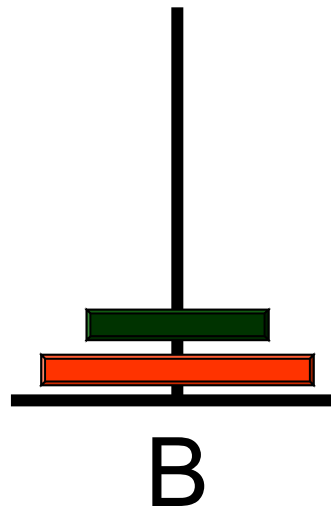
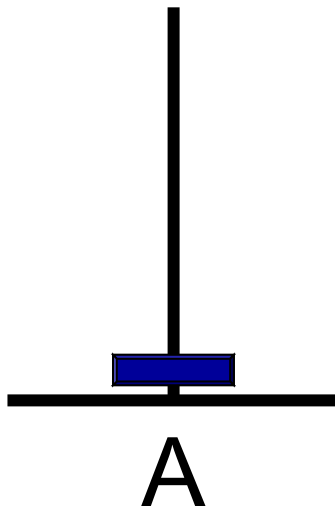
# Torre de Hanói

5	Hanoi_cham2(2, A, C, B)	
4	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
3	Hanoi_cham2(3, A, B, C)	
2	Hanoi_2(2, C, B, A)	Mover disco 2 de C para B
1	Hanoi_cham(2, C, B, A)	
0	Hanoi_1(1, C, A, B)	Mover disco 1 de C para A

**1 = Azul**

**2 = Verde**

**3 = Laranja**



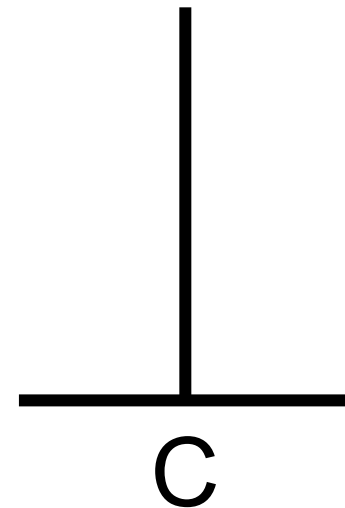
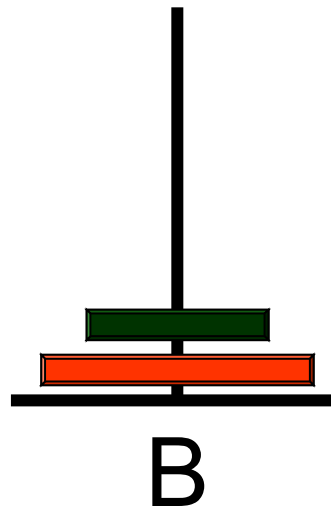
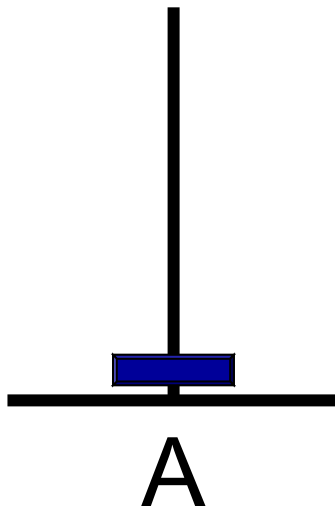
# Torre de Hanói

5	Hanoi_2(1, B, C, A)	Mover disco 1 de B para C
4	Hanoi_cham2(3, A, B, C)	
3	Hanoi_2(2, C, B, A)	Mover disco 2 de C para B
2	Hanoi_cham(2, C, B, A)	
1	Hanoi_1(1, C, A, B)	Mover disco 1 de C para A
0	Hanoi_cham2(2, C, B, A)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



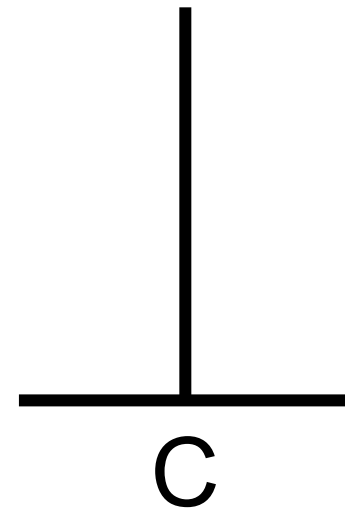
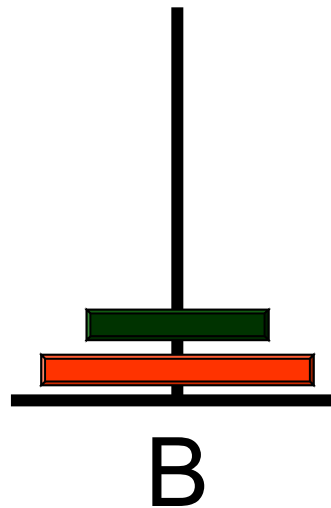
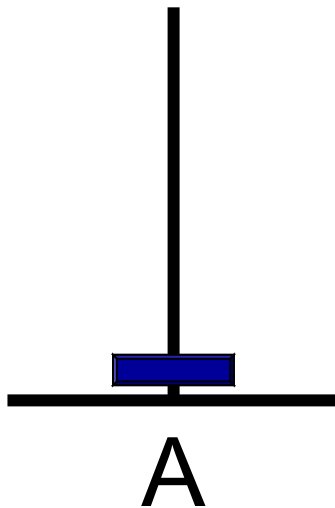
# Torre de Hanói

5	Hanoi_cham2(3, A, B, C)	
4	Hanoi_2(2, C, B, A)	Mover disco 2 de C para B
3	Hanoi_cham(2, C, B, A)	
2	Hanoi_1(1, C, A, B)	Mover disco 1 de C para A
1	Hanoi_cham2(2, C, B, A)	
0	Hanoi_2(1, A, B, C)	

**1 = Azul**

**2 = Verde**

**3 = Laranja**



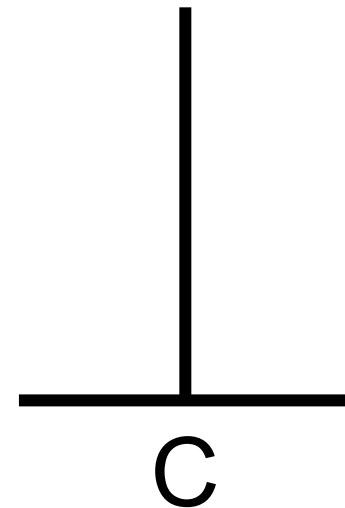
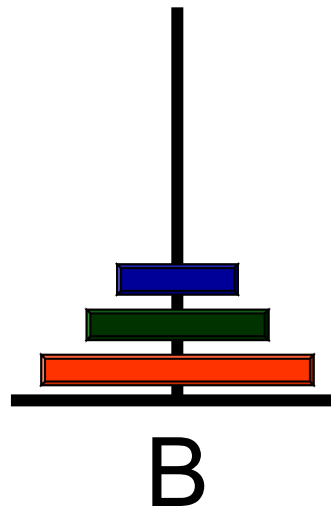
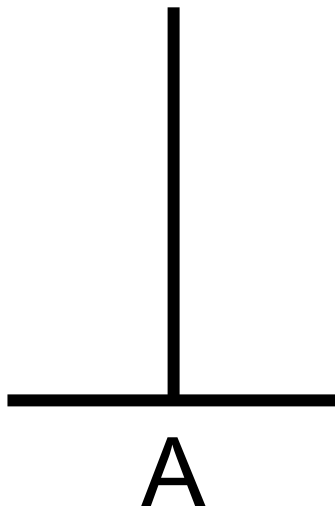
# Torre de Hanói

5	Hanoi_cham2(3, A, B, C)	
4	Hanoi_2(2, C, B, A)	Mover disco 2 de C para B
3	Hanoi_cham(2, C, B, A)	
2	Hanoi_1(1, C, A, B)	Mover disco 1 de C para A
1	Hanoi_cham2(2, C, B, A)	
0	Hanoi_2(1, A, B, C)	Mover disco 1 de A para B

**1 = Azul**

**2 = Verde**

**3 = Laranja**



# Torre de Hanói

1. Chega em 1: 3, A, B, C
2. Hanoi\_1(2, A, C, B)
3. Chega em 1: 2, A, C, B
4. Hanoi\_1(1, A, B, C)
5. Mover disco 1 de A para B
6. Mover disco 2 de A para C
7. Chega em 2: 2, A, C, B
8. Hanoi\_2(1, B, C, A)
9. Mover disco 1 de B para C
10. Mover disco 3 de A para B
11. Chega em 2: 3, A, B, C
12. Hanoi\_2(2, C, B, A)
13. Chega em 1: 2, C, B, A
14. Hanoi\_1(1, C, A, B)
15. Mover disco 1 de C para A
16. Mover disco 2 de C para B
17. Chega em 2: 2, C, B, A
18. Hanoi\_2(1, A, B, C)
19. Mover disco 1 de A para B

# Torre de Hanói (Solução Simples)

```
hanoi(disc, orig, dest, aux)
{
    if (disc == 1)
        printf("Move %d de %c para %c\n", disc, orig, dest);

    else{
        hanoi(disc-1, orig, aux, dest);
        printf("Move %d de %c para %c\n", disc, orig, dest);
        hanoi(disc-1, aux, dest, orig);
    }
}
```

# Torre de Hanói

```
hanoi(disc, orig, dest, aux)
{
    if (disc > 1) /* função n-1 de A p/ C */
        hanoi(disc-1, orig, aux, dest);

    printf("Move %d de %c para %c\n", disc, orig, dest);

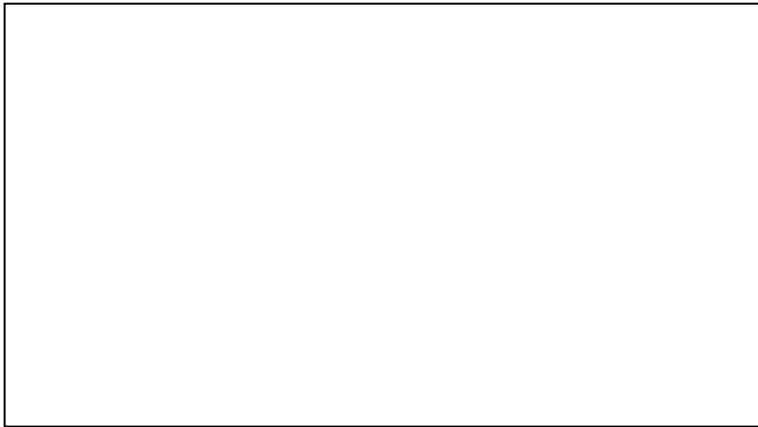
    if (disc > 1) /* função n-1 de C p/ B */
        hanoi(disc-1, aux, dest, orig);
}
```



# Execução

Utilizando as chamadas em  
memória!!!

**Chama: 3, A, B, C**



**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**FUNC\_02: 3, A, B, C**

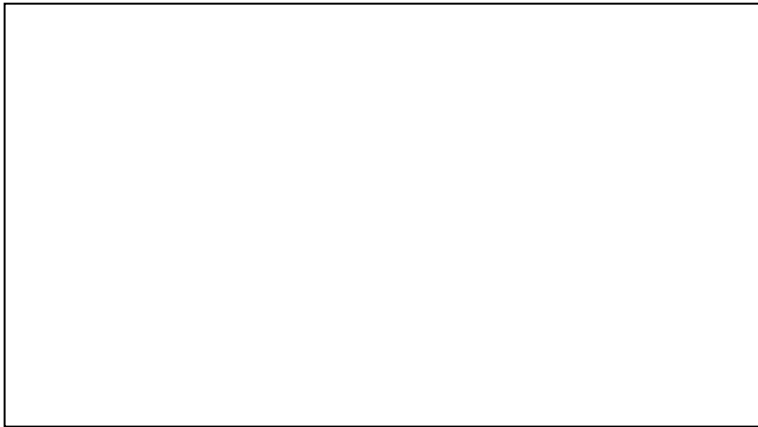


**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**FUNC\_02: 3, A, B, C**



**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**FUNC\_02: 3, A, B, C**



**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**FUNC\_02: 3, A, B, C**



**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**FUNC\_01: 2, A, C, B    FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**FUNC\_02: 3, A, B, C**

**01 - Mover disco 1 de A para B**

**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**FUNC\_02: 3, A, B, C**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**



**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**    **FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**FUNC\_02: 3, A, B, C**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**Chama: 3, A, B, C**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**FUNC\_01: 2, C, B, A    FUNC\_02: 2, C, B, A**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**FUNC\_01: 2, C, B, A    FUNC\_02: 2, C, B, A**

**Chama: 1, C, A, B**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**05 - Mover disco 1 de C para A**

**FUNC\_01: 2, C, B, A    FUNC\_02: 2, C, B, A**

**Chama: 1, C, A, B**

**05**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**06**

**FUNC\_01: 2, C, B, A**

**FUNC\_02: 2, C, B, A**

**Chama: 1, C, A, B**

**05**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**05 - Mover disco 1 de C para A**

**06 - Mover disco 2 de C para B**



**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**06**

**FUNC\_01: 2, C, B, A**

**FUNC\_02: 2, C, B, A**

**Chama: 1, C, A, B**

**05**

**Chama: 1, A, B, C**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**05 - Mover disco 1 de C para A**

**06 - Mover disco 2 de C para B**

**Chama: 3, A, B, C**

**04**

**FUNC\_01: 3, A, B, C**

**Chama: 2, A, C, B**

**02**

**FUNC\_01: 2, A, C, B**

**FUNC\_02: 2, A, C, B**

**Chama: 1, A, B, C**

**01**

**Chama: 1, B, C, A**

**03**

**FUNC\_02: 3, A, B, C**

**Chama: 2, C, B, A**

**06**

**FUNC\_01: 2, C, B, A**

**FUNC\_02: 2, C, B, A**

**Chama: 1, C, A, B**

**05**

**Chama: 1, A, B, C**

**07**

**01 - Mover disco 1 de A para B**

**02 - Mover disco 2 de A para C**

**03 - Mover disco 1 de B para C**

**04 - Mover disco 3 de A para B**

**05 - Mover disco 1 de C para A**

**06 - Mover disco 2 de C para B**

**07 - Mover disco 1 de A para B**

Fim! Ufa!