

Aula 5

Algoritmos recursivos: Torre de Hanói

Projeto e Análise de Algoritmos

Professor Eurinardo Rodrigues Costa
Universidade Federal do Ceará
Campus Russas

2021.1

Aulas Passadas

Algoritmos Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo

Aulas Passadas

PAA - Aula 5

Prof. Eurinardo

Aulas Passadas

Algoritmos

Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo

► Correção de Algoritmos Iterativos

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.
 - ▶ Exemplo: Insertion-Sort

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.
 - ▶ Exemplo: Insertion-Sort
 - ▶ Melhor caso = $an - b$.

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.
 - ▶ Exemplo: Insertion-Sort
 - ▶ Melhor caso = $an - b$.
 - ▶ Pior caso = $an^2 + bn - c$.

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.
 - ▶ Exemplo: Insertion-Sort
 - ▶ Melhor caso = $an - b$.
 - ▶ Pior caso = $an^2 + bn - c$.
- ▶ Crescimento de Funções

- ▶ Correção de Algoritmos Iterativos
 - ▶ Invariante de laço
 - ▶ Exemplo: Insertion-Sort
- ▶ Complexidade de Tempo/Espaço
 - ▶ Pior caso, Melhor caso e Caso médio.
 - ▶ Exemplo: Insertion-Sort
 - ▶ Melhor caso = $an - b$.
 - ▶ Pior caso = $an^2 + bn - c$.
- ▶ Crescimento de Funções
 - ▶ Notação O , Ω e Θ

Algoritmos Recursivos

PAA - Aula 5

Prof. Eurinardo

Aulas Passadas

Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo

Torre de Hanói

Aulas Passadas

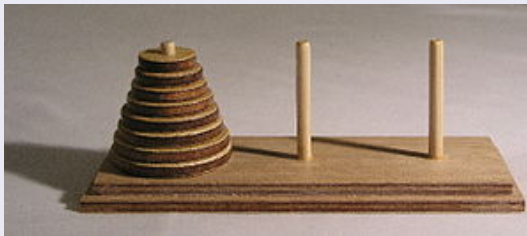
Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo



Fonte :

https://pt.wikipedia.org/wiki/Torre_de_Han%C3%B3i

Torre de Hanói

Aulas Passadas

Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo



Fonte :

https://pt.wikipedia.org/wiki/Torre_de_Han%C3%B3i

Objetivo:

Torre de Hanói

Aulas Passadas

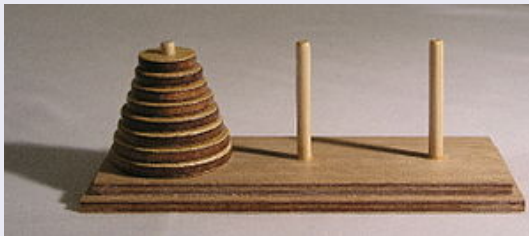
Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo



Fonte :

https://pt.wikipedia.org/wiki/Torre_de_Han%C3%B3i

Objetivo: colocar os discos do pino esquerdo no pino direito obedecendo as regras:

Torre de Hanói

Aulas Passadas

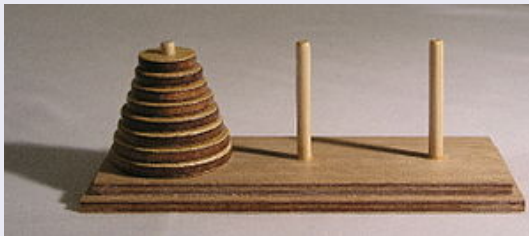
Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo



Fonte :

https://pt.wikipedia.org/wiki/Torre_de_Han%C3%B3i

Objetivo: colocar os discos do pino esquerdo no pino direito obedecendo as regras:

- mover um disco de cada vez.

Torre de Hanói

Aulas Passadas

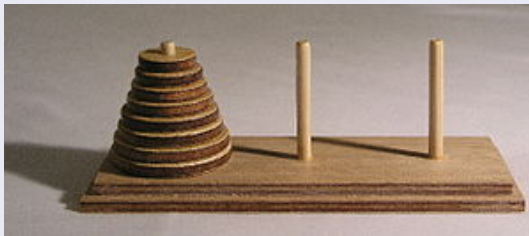
Algoritmos
Recursivos

Torre de Hanói

Ideia

Algoritmo

Análise do Tempo



Fonte :

https://pt.wikipedia.org/wiki/Torre_de_Han%C3%B3i

Objetivo: colocar os discos do pino esquerdo no pino direito obedecendo as regras:

- ▶ mover um disco de cada vez.
- ▶ um disco não pode ficar em cima de outro menor que ele.

Algoritmo 1: Hanói (n, A, B, C)

Algoritmo 2: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A
 A, B, C : pinos

Algoritmo 3: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

Algoritmo 4: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

1 **se** $n = 1$ **então**

|

Algoritmo 5: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

```
1 se  $n = 1$  então
2   | mover( $A \rightarrow C$ )
```

Algoritmo 6: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

```
1 se  $n = 1$  então
2   | mover( $A \rightarrow C$ )
3 senão
   |
```

Algoritmo 7: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

```
1 se  $n = 1$  então  
2   | mover( $A \rightarrow C$ )  
3 senão  
4   | Hanói( $n - 1, A, C, B$ )  
   |
```

Algoritmo 8: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A

A, B, C : pinos

Saída: Informar os movimentos para colocar os n
discos do pino A para o pino C

```
1 se  $n = 1$  então  
2   | mover( $A \rightarrow C$ )  
3 senão  
4   | Hanói( $n - 1, A, C, B$ )  
5   | mover( $A \rightarrow C$ )
```

Algoritmo 9: Hanói (n, A, B, C)

Entrada: n : número de disco no pino A A, B, C : pinos**Saída:** Informar os movimentos para colocar os n
discos do pino A para o pino C

```
1 se  $n = 1$  então  
2   | mover( $A \rightarrow C$ )  
3 senão  
4   | Hanói( $n - 1, A, C, B$ )  
5   | mover( $A \rightarrow C$ )  
6   | Hanói( $n - 1, B, A, C$ )  
7 fim
```



LEISERSON, C.E., STEIN, C., RIVEST, R.L.,
CORMEN T.H.

Algoritmos: teoria e prática, 3ed.

Editora Campus, ano 2012.

Obrigado!