Aula 07 – Relações de Recorrência

Norton T. Roman & Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri

2023

Custo de Algoritmos Recursivos

- Medir o número de operações de algoritmos iterativos (ou seja, sua Função de Complexidade) é razoavelmente fácil.

Custo de Algoritmos Recursivos

- Medir o número de operações de algoritmos iterativos (ou seja, sua Função de Complexidade) é razoavelmente fácil.
  - Contamos as operações desejadas, tomando cuidado com chamadas a métodos e laços

Custo de Algoritmos Recursivos

- Medir o número de operações de algoritmos iterativos (ou seja, sua Função de Complexidade) é razoavelmente fácil.
  - Contamos as operações desejadas, tomando cuidado com chamadas a métodos e laços
- Como medimos, contudo, o número de operações de algoritmos recursivos?

Custo de Algoritmos Recursivos

- Medir o número de operações de algoritmos iterativos (ou seja, sua Função de Complexidade) é razoavelmente fácil.
  - Contamos as operações desejadas, tomando cuidado com chamadas a métodos e laços
- Como medimos, contudo, o número de operações de algoritmos recursivos?
  - Via relações de recorrência

Custo de Algoritmos Recursivos

Relação de Recorrência

Uma recorrência é uma equação (ou inequação) que descreve uma função em termos de seu valor com entradas menores.

Custo de Algoritmos Recursivos

Relação de Recorrência

Uma recorrência é uma equação (ou inequação) que descreve uma função em termos de seu valor com entradas menores.

- Trata-se de um modo de definir uma função por uma expressão envolvendo a mesma função, com um ou mais valores anteriores

Custo de Algoritmos Recursivos

Relação de Recorrência

Uma recorrência é uma equação (ou inequação) que descreve uma função em termos de seu valor com entradas menores.

- Trata-se de um modo de definir uma função por uma expressão envolvendo a mesma função, com um ou mais valores anteriores
- Ou seja, uma expressão recursiva para a definição de uma função

Relações de Recorrência

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

Relações de Recorrência

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

- 1,

Relações de Recorrência

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

- 1, 4,

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

- 1, 4, 7,

Relações de Recorrência

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

- 1, 4, 7, 10,

Relações de Recorrência

Exemplo

- Quais os 5 primeiros valores da recorrência $T(n)$ abaixo?

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n-1) + 3, & \text{para } n \geq 2 \end{cases}$$

- 1, 4, 7, 10, 13

Relações de Recorrência

- Embora a relação de recorrência defina a função de custo de forma única, calcular seu resultado final é bem mais difícil

Relações de Recorrência

- Embora a relação de recorrência defina a função de custo de forma única, calcular seu resultado final é bem mais difícil
  - Se seguirmos a definição, precisaremos calcular todos os elementos na série que ela representa.

Relações de Recorrência

- Embora a relação de recorrência defina a função de custo de forma única, calcular seu resultado final é bem mais difícil
  - Se seguirmos a definição, precisaremos calcular todos os elementos na série que ela representa.
- Seria muito mais conveniente ter uma fórmula fechada para $T(n)$

Relações de Recorrência

- Embora a relação de recorrência defina a função de custo de forma única, calcular seu resultado final é bem mais difícil
  - Se seguirmos a definição, precisaremos calcular todos os elementos na série que ela representa.
- Seria muito mais conveniente ter uma fórmula fechada para $T(n)$
  - Como?

Resolvendo Relações de Recorrência

Método da Expansão

- Substituem-se os termos $T(k), k < n$, até que todos os termos $T(k), k > 1$, tenham sido substituídos por fórmulas contendo apenas $T(1)$

Resolvendo Relações de Recorrência

Método da Expansão

- Substituem-se os termos $T(k), k < n$, até que todos os termos $T(k), k > 1$, tenham sido substituídos por fórmulas contendo apenas $T(1)$
  - Nesse caso, usamos a base como 1, mas poderia ser qualquer valor

Resolvendo Relações de Recorrência

Método da Expansão

- Substituem-se os termos $T(k), k < n$, até que todos os termos $T(k), k > 1$, tenham sido substituídos por fórmulas contendo apenas $T(1)$
  - Nesse caso, usamos a base como 1, mas poderia ser qualquer valor
- Ou seja, expandimos a recorrência, de modo a obter sua solução

Resolvendo Relações de Recorrência

Método da Expansão

- Substituem-se os termos $T(k), k < n$, até que todos os termos $T(k), k > 1$, tenham sido substituídos por fórmulas contendo apenas $T(1)$
  - Nesse caso, usamos a base como 1, mas poderia ser qualquer valor
- Ou seja, expandimos a recorrência, de modo a obter sua solução
  - Essa solução é também conhecida como forma fechada da relação de recorrência

Resolvendo Relações de Recorrência

Método da Expansão

- Usa uma abordagem do tipo "expanda, suponha e verifique"

Resolvendo Relações de Recorrência

Método da Expansão

- Usa uma abordagem do tipo "expanda, suponha e verifique"
  - Usamos repetidamente a relação de recorrência para expandir a expressão para o n-ésimo termo, até que um padrão geral de comportamento da expressão possa ser induzido

Resolvendo Relações de Recorrência

Método da Expansão

- Usa uma abordagem do tipo "expanda, suponha e verifique"
  - Usamos repetidamente a relação de recorrência para expandir a expressão para o n-ésimo termo, até que um padrão geral de comportamento da expressão possa ser induzido
  - Ou seja, expandimos a relação de recorrência até que possa ser detectado o seu comportamento no caso geral

Resolvendo Relações de Recorrência

Método da Expansão

- Usa uma abordagem do tipo "expanda, suponha e verifique"
  - Usamos repetidamente a relação de recorrência para expandir a expressão para o n-ésimo termo, até que um padrão geral de comportamento da expressão possa ser induzido
  - Ou seja, expandimos a relação de recorrência até que possa ser detectado o seu comportamento no caso geral
- Finalmente, a suposição é verificada (demonstrada) por indução matemática

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:
- O que contaremos?

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:
- O que contaremos?
  - Operações aritméticas

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:
- O que contaremos?
  - Operações aritméticas
- Qual seria sua relação de recorrência?

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:
- O que contaremos?
  - Operações aritméticas

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

- Qual seria sua relação de recorrência?
  - Base:

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

- O que contaremos?
  - Operações aritméticas
- Qual seria sua relação de recorrência?
  - Base: $n = 0$, caso em que retornamos $0! = 1$

Método da Expansão

Exemplo

- Considere o algoritmo para cálculo do fatorial:

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

- O que contaremos?
  - Operações aritméticas
- Qual seria sua relação de recorrência?
  - Base: $n = 0$, caso em que retornamos $0! = 1$
  - $T(0) = 0$, pois não há operação aritmética aí

Método da Expansão

Exemplo

- Recorrência:

$T(n) =$

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 operação

$$T(n) = 1$$

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 operação mais tantas quantas forem necessárias para $n-1$

  $T(n) = 1$

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 operação mais tantas quantas forem necessárias para $n-1$

$$T(n) = 1 + T(n-1)$$

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 operação mais tantas quantas forem necessárias para $n-1$

  $T(n) = 1 + T(n-1)$

```
Entrada: inteiro n

Se n=0, retorne 1
Senão
  retorne n multiplicado pelo
  fatorial de n-1
```

- Então...

$$T(n) = \begin{cases} 0, & \text{se } n = 0 \\ T(n-1) + 1, & \text{para } n \geq 1 \end{cases}$$

Método da Expansão

Exemplo

$$T(n) = T(n-1) + 1$$

Método da Expansão

Exemplo

$$\begin{aligned} T(n) &= T(n-1)+1 \\ &= ((T(n-2)+1)+1 \end{aligned}$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(n-1) + 1 \\
&= ((T(n-2) + 1) + 1 \\
&= (((T(n-3) + 1) + 1) + 1
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(n-1)+1 \\
&= ((T(n-2)+1)+1 \\
&= (((T(n-3)+1)+1)+1 \\
&= \ldots
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(n-1) + 1 \\
&= ((T(n-2) + 1) + 1 \\
&= (((T(n-3) + 1) + 1) + 1 \\
&= \dots \\
&= (\dots((T(n-k)\underbrace{+1) + 1) + \dots + 1) + 1}_{\text{k vezes}}
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(n-1)+1 \\
&= ((T(n-2)+1)+1 \\
&= (((T(n-3)+1)+1)+1 \\
&= \ldots \\
&= (\ldots((T(n-k)\underbrace{+1)+1)+\ldots+1)+1}_{\text{k vezes}} \\
&= T(0)+\sum_{i=1}^{k} 1 \text{ (ao fim da expansão)}
\end{aligned}
$$

Método da Expansão

Exemplo

$$= \quad T(0) + k, \text{ quando } T(n-k) = T(0)$$

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(0) + k, \text{ quando } T(n-k) = T(0) \\
&= T(0) + n \text{ (pois } n - k = 0)
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(0) + k, \text{ quando } T(n-k) = T(0) \\
&= T(0) + n \text{ (pois } n - k = 0) \\
&= n \text{ (pois } T(0) = 0)
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(0) + k, \text{ quando } T(n-k) = T(0) \\
&= T(0) + n \text{ (pois } n - k = 0) \\
&= n \text{ (pois } T(0) = 0)
\end{aligned}
$$

- Então $T(n) = n$ operações aritméticas são necessárias

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(0) + k, \text{ quando } T(n-k) = T(0) \\
&= T(0) + n \text{ (pois } n - k = 0) \\
&= n \text{ (pois } T(0) = 0)
\end{aligned}
$$

- Então $T(n) = n$ operações aritméticas são necessárias
- Inferimos a forma fechada da relação de recorrência

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(0) + k, \text{ quando } T(n-k) = T(0) \\
&= T(0) + n \text{ (pois } n - k = 0) \\
&= n \text{ (pois } T(0) = 0)
\end{aligned}
$$

- Então $T(n) = n$ operações aritméticas são necessárias
- Inferimos a forma fechada da relação de recorrência
  - A resposta só virá mesmo com a demonstração de que $T(n) = n$ está correta (provando, por exemplo, por indução finita).

Método da Expansão

Exemplo

- E por que isso é uma indução?

Método da Expansão

Exemplo

- E por que isso é uma indução?
- Por conta dessa parte:

$$
\begin{aligned}
T(n) &= \ldots \\
&= (\ldots((T(n-k)\underbrace{+1)+1)+\ldots+1)+1}_{\text{k vezes}} \\
&= T(0) + \sum_{i=1}^{k} 1 \text{ (ao fim da expansão)}
\end{aligned}
$$

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?
  - Comparações

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                     compare com x

Se x=arr[meio], meio é o índice
             do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?
  - Comparações
- Qual seria sua relação de recorrência?

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?
  - Comparações
- Qual seria sua relação de recorrência?
  - Base:

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                     compare com x

Se x=arr[meio], meio é o índice
             do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?
  - Comparações
- Qual seria sua relação de recorrência?
  - Base: $|arr| = 1$, caso em que vemos se o elemento buscado é ou não o do arranjo

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                    compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Considere o algoritmo de busca binária:
- O que contaremos?
  - Comparações
- Qual seria sua relação de recorrência?
  - Base: $|arr| = 1$, caso em que vemos se o elemento buscado é ou não o do arranjo $\rightarrow T(1) = 1$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                   compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
   no subarranjo de ini a meio-1
Senão
   repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Recorrência:

$T(n) =$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
     no subarranjo de 0 a meio-1
Senão
    repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 comparação

$$T(n) = 1$$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                    compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
     no subarranjo de 0 a meio-1
Senão
    repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 comparação mais tantas quantas forem necessárias na metade inferior

$$T(n) = 2$$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
     no subarranjo de 0 a meio-1
Senão
    repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 comparação mais tantas quantas forem necessárias na metade inferior ou superior do arranjo

  $$T(n) = 2$$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
     no subarranjo de 0 a meio-1
Senão
    repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- Recorrência: Temos 1 comparação mais tantas quantas forem necessárias na metade inferior ou superior do arranjo

  $T(n) = 2 + T(\lfloor n/2 \rfloor)$, onde $n = |arr|$

```
Entrada: arranjo arr, elemento x

Se o arranjo tiver 1 elemento,
                  compare com x

Se x=arr[meio], meio é o índice
           do elemento procurado
Se x<arr[meio], repete a busca
     no subarranjo de 0 a meio-1
Senão
    repete a busca no subarranjo
         de meio+1 ao fim de arr
```

Método da Expansão

Exemplo

- E...

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(\lfloor n/2 \rfloor) + 2, & \text{para } n \geq 2 \end{cases}$$

Método da Expansão

Exemplo

- E...

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(\lfloor n/2 \rfloor) + 2, & \text{para } n \geq 2 \end{cases}$$

- Em geral, podemos ignorar pisos ($\lfloor x \rfloor$) e tetos ($\lceil x \rceil$), por se tratar de uma conta aproximada

Método da Expansão

Exemplo

- E...

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(\lfloor n/2 \rfloor) + 2, & \text{para } n \geq 2 \end{cases}$$

- Em geral, podemos ignorar pisos ($\lfloor x \rfloor$) e tetos ($\lceil x \rceil$), por se tratar de uma conta aproximada
- Então vamos expandir isso...

Método da Expansão

Exemplo

$$T(n) = T\left(\frac{n}{2}\right) + 2$$

Método da Expansão

Exemplo

$$\begin{aligned} T(n) &= T(\tfrac{n}{2}) + 2 \\ &= ((T(\tfrac{n}{4}) + 2) + 2 \end{aligned}$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(\tfrac{n}{2}) + 2 \\
&= ((T(\tfrac{n}{4}) + 2) + 2 \\
&= (((T(\tfrac{n}{8}) + 2) + 2) + 2
\end{aligned}
$$

Método da Expansão

Exemplo

$$\begin{aligned} T(n) &= T(\tfrac{n}{2}) + 2 \\ &= ((T(\tfrac{n}{4}) + 2) + 2 \\ &= (((T(\tfrac{n}{8}) + 2) + 2) + 2 \\ &= \ldots \end{aligned}$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(\tfrac{n}{2}) + 2 \\
&= ((T(\tfrac{n}{4}) + 2) + 2 \\
&= (((T(\tfrac{n}{8}) + 2) + 2) + 2 \\
&= \ldots \\
&= (\ldots((T(\tfrac{n}{2^k}) \underbrace{+2) + 2) + \ldots + 2) + 2}_{\text{k vezes}}
\end{aligned}
$$

Método da Expansão

Exemplo

$$
\begin{aligned}
T(n) &= T(\tfrac{n}{2}) + 2 \\
&= ((T(\tfrac{n}{4}) + 2) + 2 \\
&= (((T(\tfrac{n}{8}) + 2) + 2) + 2 \\
&= \dots \\
&= (\dots((T(\tfrac{n}{2^k})\underbrace{+2) + 2) + \dots + 2) + 2}_{\text{k vezes}} \\
&= T(1) + \sum_{i=1}^{k} 2 \text{ (ao fim da expansão)}
\end{aligned}
$$

Método da Expansão

Exemplo

$$= \; T(1) + 2 * k, \text{ quando } T(\tfrac{n}{2^k}) = T(1)$$

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(1) + 2 * k, \text{ quando } T(\tfrac{n}{2^k}) = T(1) \\
&= T(1) + 2 * log_2(n) \text{ (pois } \tfrac{n}{2^k} = 1)
\end{aligned}
$$

Método da Expansão

Exemplo

$$= \ T(1) + 2 * k, \text{ quando } T(\tfrac{n}{2^k}) = T(1)$$

$$= \ T(1) + 2 * log_2(n) \text{ (pois } \tfrac{n}{2^k} = 1)$$

$$= \ 1 + 2 * log_2(n) \text{ (pois } T(1) = 1)$$

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(1) + 2 * k, \text{ quando } T(\tfrac{n}{2^k}) = T(1) \\
&= T(1) + 2 * log_2(n) \text{ (pois } \tfrac{n}{2^k} = 1) \\
&= 1 + 2 * log_2(n) \text{ (pois } T(1) = 1)
\end{aligned}
$$

Então $T(n) = 2 * log_2(n) + 1$
comparações são necessárias

Método da Expansão

Exemplo

$$
\begin{aligned}
&= T(1) + 2 * k, \text{ quando } T(\tfrac{n}{2^k}) = T(1) \\
&= T(1) + 2 * log_2(n) \text{ (pois } \tfrac{n}{2^k} = 1) \\
&= 1 + 2 * log_2(n) \text{ (pois } T(1) = 1)
\end{aligned}
$$

Então $T(n) = 2 * log_2(n) + 1$ comparações são necessárias

Método da Expansão

Exemplo

- Mas isso, contudo, foi uma forma de indução lógica (embora bem baseada), não dedução

Método da Expansão

Exemplo

- Mas isso, contudo, foi uma forma de indução lógica (embora bem baseada), não dedução
- Para finalizar, temos então que demonstrar dedutivamente (via indução finita, por exemplo) que $T(n) = 2 * log_2(n) + 1$ está correta

Método da Expansão

Exemplo

- Mas isso, contudo, foi uma forma de indução lógica (embora bem baseada), não dedução
- Para finalizar, temos então que demonstrar dedutivamente (via indução finita, por exemplo) que $T(n) = 2 * log_2(n) + 1$ está correta
- Fica como exercício...

Método da Expansão

Árvore de Recorrência

- Embora a expansão possa, em tese, resolver toda relação de recorrências, há casos em que fica mais difícil
- Ex:

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ 2T(n/2) + n, & \text{para } n \geq 2 \end{cases}$$

Método da Expansão

Árvore de Recorrência

- Embora a expansão possa, em tese, resolver toda relação de recorrências, há casos em que fica mais difícil

- Ex:

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ 2T(n/2) + n, & \text{para } n \geq 2 \end{cases}$$

- Em casos assim, vale a pena construir a árvore de recorrência

Método da Expansão

Construindo a Árvore de Recorrência

- Cada nó representa o custo de um sub-problema no conjunto de invocações recursivas
- Somamos os custos em cada nível da árvore para obter um custo total do nível
  - Este é o custo adicionado pelas chamadas recursivas naquele nível
- Somamos todos os custos por nível para determinar o custo total da recursão

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n,\ T(1) = 1$

$T(n) =$

$$n$$

$$T(\tfrac{n}{2}) \qquad\qquad\qquad T(\tfrac{n}{2})$$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n,\ T(1) = 1$

$$T(n) = \quad n$$

$$T\left(\tfrac{n}{2}\right) \qquad\qquad T\left(\tfrac{n}{2}\right)$$

Custo adicionado
no nível mais alto

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = \color{red}{2T(n/2)} + n, \; T(1) = 1$

$T(n) =$

$$n$$

$$T\left(\frac{n}{2}\right) \qquad\qquad T\left(\frac{n}{2}\right)$$

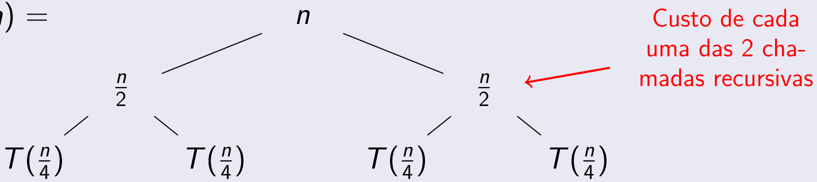
Custo de cada uma das 2 chamadas recursivas

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n,\ T(1) = 1$

$T(n) =$

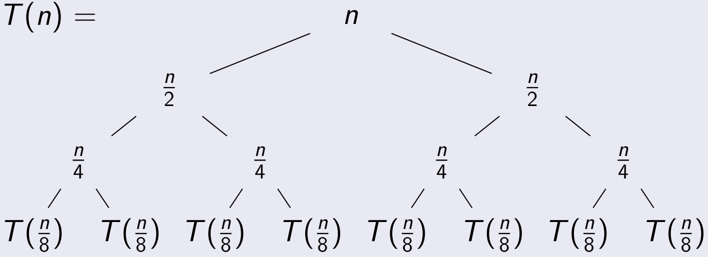


Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

$T(n) =$



Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

$T(n) =$

$$n$$

$$\frac{n}{2} \qquad \frac{n}{2}$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4}$$

$$T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8})$$

$$\cdots$$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n,\ T(1) = 1$

$T(n) =$ n $= n$

$\frac{n}{2}$ $\frac{n}{2}$

$\frac{n}{4}$ $\frac{n}{4}$ $\frac{n}{4}$ $\frac{n}{4}$

$T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$ $T(\frac{n}{8})$

$\cdots$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

$$T(n) = \qquad n \qquad = n$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad = 2 \times \frac{n}{2} = n$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4}$$

$$T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right)$$

$$\ldots$$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n,\ T(1) = 1$

$T(n) =$

$$n \qquad = n$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad = 2 \times \frac{n}{2} = n$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad = 4 \times \frac{n}{4} = n$$

$$T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8})$$

$$\ldots$$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

$$T(n) = \qquad n \qquad = n$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad = 2 \times \frac{n}{2} = n$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad = 4 \times \frac{n}{4} = n$$

$$T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \qquad = 8 \times \frac{n}{8} = n$$

$$\cdots$$

Método da Expansão

Árvore de Recorrência – Exemplo

- $T(n) = 2T(n/2) + n, \; T(1) = 1$

$$
\begin{array}{lcccccccccccccccl}
T(n) = & & & & & & & n & & & & & & & & & = n \\
& & & \frac{n}{2} & & & & & & & & \frac{n}{2} & & & & & = 2 \times \frac{n}{2} = n \\
& & \frac{n}{4} & & \frac{n}{4} & & & & & & \frac{n}{4} & & \frac{n}{4} & & & & = 4 \times \frac{n}{4} = n \\
& T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & & & & & & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & & & = 8 \times \frac{n}{8} = n \\
& & & & & & & \ldots & & & & & & & & & \ldots
\end{array}
$$

Método da Expansão

Árvore de Recorrência – Exemplo

$$T(n) = \qquad n \qquad = n$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad = 2 \times \frac{n}{2} = n$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad = 4 \times \frac{n}{4} = n$$

$$T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \quad T(\tfrac{n}{8}) \qquad = 8 \times \frac{n}{8} = n$$

$$\ldots \qquad \ldots$$

- Note que cada nível da recorrência acrescenta n ao resultado final.

Método da Expansão

Árvore de Recorrência – Exemplo

$$
\begin{array}{lccccccccl}
T(n) = & & & & n & & & & & = n \\
& & \frac{n}{2} & & & & \frac{n}{2} & & & = 2 \times \frac{n}{2} = n \\
& \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & & = 4 \times \frac{n}{4} = n \\
& T(\frac{n}{8})\ T(\frac{n}{8}) & & T(\frac{n}{8})\ T(\frac{n}{8}) & & T(\frac{n}{8})\ T(\frac{n}{8}) & & T(\frac{n}{8})\ T(\frac{n}{8}) & & = 8 \times \frac{n}{8} = n \\
& & & & \dots & & & & & \dots
\end{array}
$$

- Note que cada nível da recorrência acrescenta n ao resultado final. Mas, quantos níveis há?

Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:

$T(n) =$

$$n$$

$$\frac{n}{2} \qquad \frac{n}{2}$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4}$$

$$T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right)$$

$$\vdots \quad \vdots \qquad \cdots \qquad \vdots \quad \vdots$$

$$T(1) \quad T(1) \qquad \cdots \qquad T(1) \quad T(1)$$

Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:

$T(n) =$ n $\qquad T(\frac{n}{2^0})$

$\frac{n}{2} \qquad \frac{n}{2}$

$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4}$

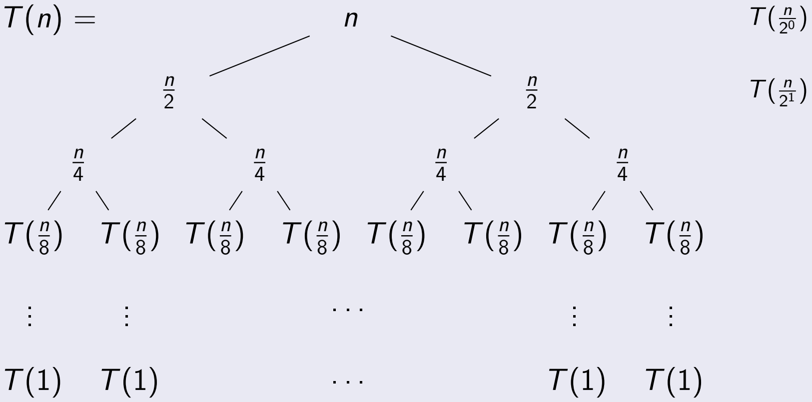
$T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8})$

$\vdots \qquad \vdots \qquad \cdots \qquad \vdots \qquad \vdots$

$T(1) \quad T(1) \qquad \cdots \qquad T(1) \quad T(1)$

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:



Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:

$$T(n) = \qquad n \qquad T\left(\frac{n}{2^0}\right)$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad T\left(\frac{n}{2^1}\right)$$

$$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad T\left(\frac{n}{2^2}\right)$$

$$T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right) \quad T\left(\frac{n}{8}\right)$$

$$\vdots \qquad \vdots \qquad \cdots \qquad \vdots \qquad \vdots$$

$$T(1) \qquad T(1) \qquad \cdots \qquad T(1) \qquad T(1)$$

Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:

$$T(n) = \quad n \qquad\qquad T\left(\frac{n}{2^0}\right)$$

$$\frac{n}{2} \qquad \frac{n}{2} \qquad\qquad T\left(\frac{n}{2^1}\right)$$

$$\frac{n}{4} \quad \frac{n}{4} \quad \frac{n}{4} \quad \frac{n}{4} \qquad\qquad T\left(\frac{n}{2^2}\right)$$

$$T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \; T\left(\frac{n}{8}\right) \qquad T\left(\frac{n}{2^3}\right)$$

$$\vdots \quad \vdots \quad \cdots \quad \vdots \quad \vdots$$

$$T(1) \quad T(1) \quad \cdots \quad T(1) \quad T(1)$$

Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:

$T(n) =$ $\qquad n \qquad T(\frac{n}{2^0})$

$\frac{n}{2} \qquad \frac{n}{2} \qquad T(\frac{n}{2^1})$

$\frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad \frac{n}{4} \qquad T(\frac{n}{2^2})$

$T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \quad T(\frac{n}{8}) \qquad T(\frac{n}{2^3})$

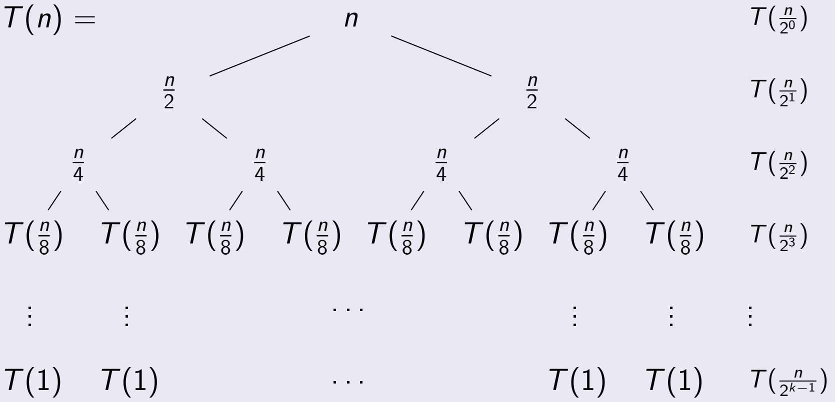
$\vdots \quad \vdots \quad \cdots \quad \vdots \quad \vdots \quad \vdots$

$T(1) \quad T(1) \quad \cdots \quad T(1) \quad T(1)$

Método da Expansão

Árvore de Recorrência – Exemplo

- Vejamos como a recorrência reduz a cada nível:



Método da Expansão

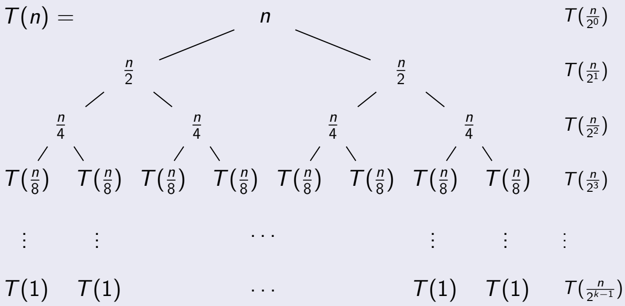
Árvore de Recorrência – Exemplo

$$
\begin{array}{lccccccccl}
T(n) = & & & & n & & & & & T(\frac{n}{2^0}) \\
& & \frac{n}{2} & & & & \frac{n}{2} & & & T(\frac{n}{2^1}) \\
& \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & & T(\frac{n}{2^2}) \\
T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & & T(\frac{n}{2^3}) \\
\vdots & \vdots & & & \cdots & & \vdots & \vdots & & \vdots \\
T(1) & T(1) & & & \cdots & & T(1) & T(1) & & T(\frac{n}{2^{k-1}})
\end{array}
$$

- Ou seja, temos k níveis, sendo que, no k-ésimo nível, $\frac{n}{2^{k-1}} = 1 \Rightarrow k = log_2(n) + 1$

Método da Expansão

Árvore de Recorrência – Exemplo



- Com $log_2(n) + 1$ níveis, e n operações por nível, temos um total de $nlog_2(n) + n$ operações.

Método da Expansão

Árvore de Recorrência – Exemplo

$$
\begin{array}{ccccccccc}
T(n) = & & & & n & & & & T(\frac{n}{2^0}) \\
& & \frac{n}{2} & & & & \frac{n}{2} & & T(\frac{n}{2^1}) \\
& \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & & \frac{n}{4} & T(\frac{n}{2^2}) \\
T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{8}) & T(\frac{n}{2^3}) \\
\vdots & \vdots & & \cdots & & & \vdots & \vdots & \vdots \\
T(1) & T(1) & & \cdots & & & T(1) & T(1) & T(\frac{n}{2^{k-1}})
\end{array}
$$

- Mais uma vez, temos que demonstrar via indução que nossa resposta está correta.

Resolvendo Relações de Recorrência

Método da Substituição

- O método da expansão é bastante útil

Resolvendo Relações de Recorrência

Método da Substituição

- O método da expansão é bastante útil
  - Muitas vezes, contudo, não é fácil chegar a uma forma fechada para a relação de recorrência

Resolvendo Relações de Recorrência

Método da Substituição

- O método da expansão é bastante útil
  - Muitas vezes, contudo, não é fácil chegar a uma forma fechada para a relação de recorrência
- Que fazer?

Resolvendo Relações de Recorrência

Método da Substituição

- O método da expansão é bastante útil
  - Muitas vezes, contudo, não é fácil chegar a uma forma fechada para a relação de recorrência
- Que fazer?

Resolvendo Relações de Recorrência

Método da Substituição

- O método da expansão é bastante útil
  - Muitas vezes, contudo, não é fácil chegar a uma forma fechada para a relação de recorrência
- Que fazer?
- Método da substituição

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:
  - Pressupor a forma da solução

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:
  - Pressupor a forma da solução
  - Usar indução finita para encontrar constantes e mostrar que a solução funciona

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:
  - Pressupor a forma da solução
  - Usar indução finita para encontrar constantes e mostrar que a solução funciona
- Aplicável quando podemos ver a forma da resposta

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:
  - Pressupor a forma da solução
  - Usar indução finita para encontrar constantes e mostrar que a solução funciona
- Aplicável quando podemos ver a forma da resposta
  - Como quando a recorrência é muito parecida com outra que já foi vista

Resolvendo Relações de Recorrência

Método da Substituição

- Consiste de dois passos:
  - Pressupor a forma da solução
  - Usar indução finita para encontrar constantes e mostrar que a solução funciona
- Aplicável quando podemos ver a forma da resposta
  - Como quando a recorrência é muito parecida com outra que já foi vista
  - É mais comumente usada para estabelecer limites superiores e inferiores para uma recorrência

Resolvendo Relações de Recorrência

Método da Substituição

- Uma vez que estamos supondo a solução, podemos adicionar constantes a ela

Resolvendo Relações de Recorrência

Método da Substituição

- Uma vez que estamos supondo a solução, podemos adicionar constantes a ela
- Ex:
  - $T(n) = c_1 n * log_2(n) + c_2 n$, em vez de $T(n) = n * log_2(n) + n$, ou, tipicamente, $T(n) \in O(n * log_2(n))$

Resolvendo Relações de Recorrência

Método da Substituição

- Uma vez que estamos supondo a solução, podemos adicionar constantes a ela
- Ex:
  - $T(n) = c_1 n * log_2(n) + c_2 n$, em vez de $T(n) = n * log_2(n) + n$, ou, tipicamente, $T(n) \in O(n * log_2(n))$
- Essas constantes são então definidas durante a prova

Resolvendo Relações de Recorrência

Método da Substituição

- Uma vez que estamos supondo a solução, podemos adicionar constantes a ela
- Ex:
  - $T(n) = c_1 n * log_2(n) + c_2 n$, em vez de $T(n) = n * log_2(n) + n$, ou, tipicamente, $T(n) \in O(n * log_2(n))$
- Essas constantes são então definidas durante a prova
  - Ou seja, definidas de modo a fazer com que a prova dê certo.

Método da Substituição

Exemplo

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n/2) + 2, & \text{para } n \geq 2 \end{cases}$$

Método da Substituição

Exemplo

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n/2) + 2, & \text{para } n \geq 2 \end{cases}$$

Vamos “chutar” que $T(n) \in O(log_2(n))$

Método da Substituição

Exemplo

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n/2) + 2, & \text{para } n \geq 2 \end{cases}$$

Vamos "chutar" que $T(n) \in O(log_2(n))$

Ou seja, existem constantes positivas c e m, de forma que $0 \leq T(n) \leq c * log_2(n)$ para todo $n \geq m$.

Método da Substituição

Exemplo

$$T(n) = \begin{cases} 1, & \text{se } n = 1. \\ T(n/2) + 2, & \text{para } n \geq 2 \end{cases}$$

Vamos "chutar" que $T(n) \in O(log_2(n))$

Ou seja, existem constantes positivas c e m, de forma que $0 \leq T(n) \leq c * log_2(n)$ para todo $n \geq m$.

Lembrete: Vamos provar por indução finita, pois não temos a equação fechada de T(n).

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n) \quad \text{para } n = 1$$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n) \qquad \text{para } n = 1$$
$$1 \leq c * log_2(1)$$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n) \qquad \text{para } n = 1$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

para $n = 1$
Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

para $n = 1$
Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

$$T(n) \leq c * log_2(n)$$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$

para $n = 1$

$$1 \leq c * log_2(1)$$

$$1 \leq 0$$

Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

$$T(n) \leq c * log_2(n)$$

para $n = 2$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

para $n = 1$
Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

$$T(n) \leq c * log_2(n)$$
$$T(2) \leq c * log_2(2)$$

para $n = 2$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

para $n = 1$
Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

$$T(n) \leq c * log_2(n)$$
$$T(2) \leq c * log_2(2)$$
$$3 \leq c * 1$$

para $n = 2$

Método da Substituição

Exemplo

Passo base:

$$T(n) \leq c * log_2(n)$$
$$1 \leq c * log_2(1)$$
$$1 \leq 0$$

para $n = 1$
Não funcionou, mas não precisa funcionar para $n = 1$, mas a partir de um dado n.

$$T(n) \leq c * log_2(n)$$
$$T(2) \leq c * log_2(2)$$
$$3 \leq c * 1$$

para $n = 2$

válido para $c \geq 3$

Método da Substituição

Exemplo

Hipótese de Indução:

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Passo Indutivo:

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

Passo Indutivo:

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$

$T(n) \leq c * (log_2(n) - log_2 2) + 2$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$

$T(n) \leq c * (log_2(n) - log_2 2) + 2$

$T(n) \leq c * (log_2(n) - 1) + 2$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$

$T(n) \leq c * (log_2(n) - log_2 2) + 2$

$T(n) \leq c * (log_2(n) - 1) + 2$

$T(n) \leq c * log_2(n) - c + 2$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$
$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$
$T(n) \leq c * (log_2(n) - log_2 2) + 2$
$T(n) \leq c * (log_2(n) - 1) + 2$
$T(n) \leq c * log_2(n) - c + 2$

Nossa hipótese é que $T(n) \leq c * log_2(n)$, o que será verdade para $c \geq 2$

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$
$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$
$T(n) \leq c * (log_2(n) - log_2 2) + 2$
$T(n) \leq c * (log_2(n) - 1) + 2$
$T(n) \leq c * log_2(n) - c + 2$

Nossa hipótese é que $T(n) \leq c * log_2(n)$, o que será verdade para $c \geq 2$
Lembrando que $c \geq 3$ (passo base)

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$
$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$
$T(n) \leq c * (log_2(n) - log_2 2) + 2$
$T(n) \leq c * (log_2(n) - 1) + 2$
$T(n) \leq c * log_2(n) - c + 2$
$T(n) \leq c * log_2(n) - 3 + 2$

Nossa hipótese é que $T(n) \leq c * log_2(n)$, o que será verdade para $c \geq 2$
Lembrando que $c \geq 3$ (passo base)

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$

$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$

$T(n) \leq c * (log_2(n) - log_2 2) + 2$

$T(n) \leq c * (log_2(n) - 1) + 2$

$T(n) \leq c * log_2(n) - c + 2$

$T(n) \leq c * log_2(n) - 3 + 2$

$T(n) \leq c * log_2(n) - 1$

Nossa hipótese é que $T(n) \leq c * log_2(n)$, o que será verdade para $c \geq 2$

Lembrando que $c \geq 3$ (passo base)

Método da Substituição

Exemplo

Hipótese de Indução:

$T(n/2) \leq c * log_2(n/2)$

Para $T(n) = T(n/2) + 2$
$T(n/2) = T(n) - 2$

Passo Indutivo:

$T(n) - 2 \leq c * log_2(n/2)$
$T(n) \leq c * (log_2(n) - log_2 2) + 2$
$T(n) \leq c * (log_2(n) - 1) + 2$
$T(n) \leq c * log_2(n) - c + 2$
$T(n) \leq c * log_2(n) - 3 + 2$
$T(n) \leq c * log_2(n) - 1$

Nossa hipótese é que $T(n) \leq c * log_2(n)$, o que será verdade para $c \geq 2$
Lembrando que $c \geq 3$ (passo base)

Provamos que $T(n) \in O(log_2(n))$

Referências

- Ziviani, Nivio. Projeto de Algoritmos: com implementações em Java e C++. Cengage. 2007.
- Cormen, Thomas H., Leiserson, Charles E., Rivest, Ronald L., Stein, Clifford. Introduction to Algorithms. 2a ed. MIT Press, 2001.
- Gersting, Judith L. Fundamentos Matemáticos para a Ciência da Computação. 3a ed. LTC. 1993.

Aula 07 – Relações de Recorrência

Norton T. Roman & Luciano A. Digiampietri
digiampietri@usp.br
@digiampietri

2023