

Solução de Recorrências: método da Substituição

- consiste em provar por indução que $T(n)$ é $O(f(n))$ e/ou $\Omega(f(n))$ ($T(n) \leq c \cdot f(n)$ ou $T(n) \geq c \cdot f(n)$).
- ↳ Então $f(n)$ precisa ser conhecida (p.ex. um dos outros métodos podem ser usados).
- É preciso provar, no passo da indução, **exatamente** o que foi proposto.
- ↳ Ex: para provar que $T(n)$ é $O(f(n))$, podemos provar $T(n) \leq c f(n)$
não estará certo se no passo provarmos $T(n) \leq 2c f(n)$
 $T(n) \leq f(n) + 10$ ou $T(n) \leq (c+1)f(n)$

Método de substituição: exemplo 1

- Resolva a recorrência $T(n) = 2T(\frac{n}{2}) + \theta(n) \Rightarrow T(n) = 2T(\frac{n}{2}) + n$
Chute: $T(n)$ é $O(n^2)$
Vamos provar, por indução em n , que $T(n) \leq c \cdot n^2$.

BASE: $n=1$. Sabemos que $T(1)=1$ e $c \cdot 1^2 = c$. Então vale se $c \geq 1$.

HIP: $T(k) \leq c \cdot k^2 \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$. Sabemos que $T(n) = 2T(\frac{n}{2}) + n$. Por HI, como $\frac{n}{2} < n$ (já que $n > 1$), vale que $T(\frac{n}{2}) \leq c \cdot (\frac{n}{2})^2 = \frac{cn^2}{4}$.

Então

$$T(n) = 2T(\frac{n}{2}) + n \leq 2\left(c \frac{n^2}{4}\right) + n = \frac{1}{2}cn^2 + n \leq cn^2$$

onde a última expressão vale se $\frac{1}{2}cn^2 + n \leq cn^2$, que vale se $n \leq \frac{cn^2}{2}$, ou $\frac{cn}{2} \geq 1$, ou $c \geq \frac{2}{n}$.

Como $\frac{2}{n} < 2$ sempre que $n > 1$, basta que $c \geq 2$.

No caso base precisamos que $c \geq 1$, então tomamos $c=2$ e $n_0=1$.
Provamos que $T(n) \leq 2n^2$ e $\therefore T(n)$ é $O(n^2)$.

QED

Podíamos ter provado que $T(n) \leq cn^2 + dn$, o que também nos mostraria que $T(n)$ é $O(n^2)$. Poderia tentar provar algum resultado com θ para $T(n)$.

Método da substituição : exemplo 2

→ Resolva a recorrência $T(n) \leq 3T(\frac{n}{3}) + 1$

Cluta: $T(n)$ é $O(n)$

Vamos provar, por indução em n , que $T(n) \leq c \cdot n$ para todo n .

BASE: $n=1$. Temos $T(1)=1$ e $c \cdot 1 = c$. Então vale se $c \geq 1$.

HIP: $T(k) \leq c \cdot k \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$ e sabemos que $T(n) \leq 3T(\frac{n}{3}) + 1$.

Como $\frac{n}{3} < n$ quando $n > 1$, vale por HI que $T(\frac{n}{3}) \leq c \cdot \frac{n}{3}$.

Então $T(n) = 3T(\frac{n}{3}) + 1 \leq 3(c \cdot \frac{n}{3}) + 1 = cn + 1$. mas $cn + 1$ nunca será $\leq cn$.

FALHOU!

Vamos provar, por indução em n , que $T(n) \leq cn + d$ para todo n .

BASE: $n=1$. Temos $T(1)=1$ e $c \cdot 1 + d = c + d$. Então vale se $c + d \geq 1$.

HIP: $T(k) \leq ck + d \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$ e sabemos que $T(n) \leq 3T(\frac{n}{3}) + 1$.

Como $\frac{n}{3} < n$ quando $n > 1$, vale por HI que $T(\frac{n}{3}) \leq c \cdot \frac{n}{3} + d$.

Então $T(n) = 3T(\frac{n}{3}) + 1 \leq 3(c \cdot \frac{n}{3} + d) + 1 = cn + 3d + 1$

E $cn + 3d + 1 \leq cn + d$ sempre que $3d + 1 \leq d$, ou $2d \leq -1$, ou $d \leq -\frac{1}{2}$.

Precisamos então que $c + d \geq 1$ e $d \leq -\frac{1}{2}$. Tome $d = -\frac{1}{2}$ e $c = 2$.

Provamos que $T(n) \leq 2n - \frac{1}{2} \therefore T(n)$ é $O(n)$.

CQD

Método da substituição: voltando ao exemplo 1

→ Resolva a recorrência $T(n) = 2T(\frac{n}{2}) + n$

Chute: $T(n)$ é $\Omega(n^2)$

Vamos provar, por indução em n , que $T(n) \geq c \cdot n^2$.

BASE: $n=1$. Temos $T(1)=1$ e $c \cdot 1^2 = c$. Então vale se $c \leq 1$.

HIP: $T(k) \geq c \cdot k^2 \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$ e $T(n) = 2T(\frac{n}{2}) + n$.

Como $\frac{n}{2} < n$ se $n > 1$, então por HI vale que $T(\frac{n}{2}) \geq c \cdot (\frac{n}{2})^2$.

Então $T(n) = 2T(\frac{n}{2}) + n \geq 2(c \cdot \frac{n^2}{4}) + n = \frac{1}{2}cn^2 + n$.

Deveríamos ter $\frac{1}{2}cn^2 + n \geq cn^2$ para valer, mas isso claramente é impossível.

CONCLUSÃO: não conseguimos provar que $T(n)$ é $\Omega(n^2)$.

mas não provamos que não é $\Omega(n^2)$!

Acontece que $T(n)$ de fato não é $\Omega(n^2)$. Conseguiremos provar que ela é $\Omega(n \log n)$. Sabendo disso, poderemos afirmar que não é $\Omega(n^2)$.

Vamos também tentar melhorar o limitante superior!

Método de substituição: voltando ao exemplo 1

→ Resolva a recorrência $T(n) = 2T(\frac{n}{2}) + n$

Chute: $T(n)$ é $O(n \log n)$

Vamos provar, por indução em n , que $T(n) \leq c \cdot n \log n$

BASE: $n=1$. Sabemos que $T(1)=1$ e $c \cdot 1 \log 1 = 0$. mas $1 \neq 0$!!

FALHOU NO CASO BASE!

BASE: $n=2$. Sabemos que $T(2) = 2 \cdot T(1) + 2 = 4$ e $c \cdot 2 \cdot \log 2 = 2c$.

Então vale se $4 \leq 2c$, ou $c \geq 2$

HIP: $T(k) \leq c \cdot k \log k \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$. Sabemos que $T(n) = 2T(\frac{n}{2}) + n$. Por HI,
como $\frac{n}{2} < n$ (já que $n > 1$), vale que $T(\frac{n}{2}) \leq c \cdot \frac{n}{2} \cdot \log \frac{n}{2} =$
 $= \frac{1}{2} c n (\log n - \log 2) = \frac{1}{2} c n \log n - \frac{1}{2} c n$. Então

$$T(n) = 2T(\frac{n}{2}) + n \leq 2 \left(\frac{1}{2} c n \log n - \frac{1}{2} c n \right) + n = c n \log n - c n + n$$

É $c n \log n - c n + n \leq c n \log n$ se $-c n + n \leq 0$, ou $c n \geq n$,
ou $c \geq 1$.

No caso base precisamos que $c \geq 2$, então tomamos $c=2$ e $n_0=2$.
Provamos que $T(n) \leq 2n \log n$ e $\therefore T(n)$ é $O(n \log n)$.

CQD

Método de substituição: voltando ao exemplo 1

→ Resolva a recorrência $T(n) = 2T(\frac{n}{2}) + n$

Chute: $T(n)$ é $\Omega(n \log n)$

Vamos provar, por indução em n , que $T(n) \geq c \cdot n \log n$.

BASE: $n=1$. Sabemos que $T(1)=1$ e $c \cdot 1 \log 1 = 0$. Então vale se $1 \geq 0$.

HIP: $T(k) \geq c \cdot k \log k \quad \forall 1 \leq k < n$.

PASSO: Seja $n > 1$. Sabemos que $T(n) = 2T(\frac{n}{2}) + n$. Por HI, como $\frac{n}{2} < n$ (já que $n > 1$), vale que $T(\frac{n}{2}) \geq c \cdot \frac{n}{2} \cdot \log \frac{n}{2} =$
 $= \frac{1}{2} c n (\log n - \log 2) = \frac{1}{2} c n \log n - \frac{1}{2} c n$. Então

$$T(n) = 2T(\frac{n}{2}) + n \geq 2 \left(\frac{1}{2} c n \log n - \frac{1}{2} c n \right) + n = c n \log n - c n + n$$

 $\geq c n \log n - c n + n \geq c n \log n$ se $-c n + n \geq 0$, ou $c n \leq n$,
ou $c \leq 1$.

Então tomamos $c=1$ e $n_0=1$.

Provamos que $T(n) \geq n \log n$ e $\therefore T(n)$ é $\Omega(n \log n)$.
CQD

Esses dois últimos resultados mostram que $T(n)$ é $\theta(n \log n)$.

Soluções de Recorrências: método da iteração

→ consiste em expandir a recorrência até chegar no caso base, que sabemos calcular diretamente.

Método da iteração: exemplo 1

→ Resolva a recorrência $T(n) \leq T\left(\frac{n}{2}\right) + 1$

$$T(n) \leq T\left(\frac{n}{2}\right) + 1$$

$$T(n) \leq T\left(\frac{n}{4}\right) + 1 + 1 = T\left(\frac{n}{4}\right) + 2$$

$$T(n) \leq T\left(\frac{n}{8}\right) + 1 + 2 = T\left(\frac{n}{8}\right) + 3$$

⋮

$$T(n) \leq T\left(\frac{n}{2^i}\right) + i$$

Como $\frac{n}{2^i} = 1$ quando $i = \lg n$,

$$T(n) \leq T(1) + \lg n = 1 + \lg n$$

Então $T(n)$ é $O(\lg n)$.

Método da iteração: exemplo 2

→ Resolva a recorrência $T(n) = 2T(n-1) + n$

$$T(n) = 2T(n-1) + n$$

$$T(n) = 2[2T(n-2) + n-1] + n = 4T(n-2) + 2(n-1) + n$$

$$T(n) = 4[2T(n-3) + n-2] + 2(n-1) + n = 8T(n-3) + 4(n-2) + 2(n-1) + n$$

⋮

$$T(n) = 2^i T(n-i) + \sum_{j=0}^{i-1} 2^j (n-j)$$

Como $n-i=1$ quando $i=n-1$, temos

$$T(n) = 2^{n-1} T(1) + \sum_{j=0}^{n-2} 2^j (n-j) = 2^{n-1} + \underbrace{\sum_{j=0}^{n-2} 2^j n - \sum_{j=0}^{n-2} 2^j j}_{\star}$$

Tendo $T(n) = \star$, temos duas opções:

- 1) mostrar que $c_1 \cdot \omega \leq \star \leq c_2 \cdot \omega \Rightarrow T(n) \in \Theta(\omega)$
- 2) mostrar que $\star \leq c \cdot \omega \Rightarrow T(n) \in O(\omega)$.

2) Como $2^{n-1} + n \left(\frac{2^{n-1}-1}{2-1} \right) - \sum_{j=0}^{n-2} 2^j j \leq 2^{n-1} + n(2^{n-1}-1)$
e $2^{n-1} + n(2^{n-1}-1) = \frac{1}{2} 2^n + n \cdot \frac{1}{2} 2^n - n \leq \frac{1}{2} \cdot 2^n + \frac{1}{2} n 2^n$,
vemos que $T(n) \in O(n 2^n)$

1) Como $\sum_{j=0}^{n-2} 2^j j = \frac{1}{2} (2^n n - 3 \cdot 2^n + 4)$, temos que

$$2^{n-1} + n \left(\frac{2^{n-1}-1}{2-1} \right) - \sum_{j=0}^{n-2} 2^j j = \frac{1}{2} \cdot 2^n + \frac{1}{2} n 2^n - n - \frac{1}{2} n 2^n + \frac{3}{2} \cdot 2^n - 2$$
$$= 2 \cdot 2^n - n - 2, \text{ de onde vemos que } T(n) \in \Theta(2^n)$$

Método da iteração: exemplo 3

→ Resolva a recorrência $T(n) = 2T(\frac{n}{3}) + n$

$$T(n) = 2T(\frac{n}{3}) + n$$

$$T(n) = 2[2T(\frac{n}{9}) + \frac{n}{3}] + n = 4T(\frac{n}{9}) + \frac{2}{3}n + n$$

$$T(n) = 4[2T(\frac{n}{27}) + \frac{n}{9}] + \frac{2}{3}n + n = 8T(\frac{n}{27}) + \frac{4}{9}n + \frac{2}{3}n + n$$

⋮

$$T(n) = 2^i T(\frac{n}{3^i}) + \sum_{j=0}^{i-1} (\frac{2}{3})^j n$$

Como $\frac{n}{3^i} = 1$ quando $i = \log_3 n$, temos

$$T(n) = 2^{\log_3 n} \cdot T(1) + \sum_{j=0}^{\log_3 n - 1} (\frac{2}{3})^j n = n^{\log_3 2} + n \left(\sum_{j=0}^{\log_3 n - 1} (\frac{2}{3})^j \right)$$

$$= n^{\log_3 2} + n \left(\frac{1 - (\frac{2}{3})^{\log_3 n}}{1 - \frac{2}{3}} \right)$$

$$= n^{\log_3 2} + 3n - 3n \cdot n^{\log_3 2/3}$$

$$= n^{\log_3 2} + 3n - 3n^{(1 + \log_3 2 - \log_3 3)}$$

$$= n^{\log_3 2} + 3n - 3n^{\log_3 2}$$

$$= 3n - 2n^{\log_3 2}$$

(note que $\log_3 2 < \log_3 3 = 1$)

Então $T(n)$ é $\Theta(n)$.