Análise de Algoritmos

Essas transparências foram adaptadas das transparências do Prof. Paulo Feofiloff e do Prof. José Coelho de Pina.

Resolução de recorrências

- Substitua a notação assintótica por função da classe.
- Restrinja-se a n potência de algo, se necessário.
- Estipule que na base o valor é 1.
- Use expansão ou árvore de recorrência para determinar um “chute” de solução.
- Confira se o chute está correto.

Resolução de recorrências

- Substitua a notação assintótica por função da classe.
- Restrinja-se a n potência de algo, se necessário.
- Estipule que na base o valor é 1.
- Use expansão ou árvore de recorrência para determinar um “chute” de solução.
- Confira se o chute está correto.

Exemplos:

- $T(n) \;=\; T(\lfloor n/2 \rfloor) + \Theta(1)$
- $T(n) \;=\; T(n-1) + \Theta(n)$
- $T(n) \;=\; 3T(\lfloor n/2 \rfloor) + \Theta(n)$
- $T(n) \;=\; 2T(\lfloor n/2 \rfloor) + \Theta(n^2)$

Resolução de recorrências

Recorrência do Mergesort:

$$T(n) \;=\; T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n)$$

Resolução de recorrências

Recorrência do Mergesort:

$$T(n) \ = \ T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$ e $T(n) = 2\,T(n/2) + n$ para $n \geq 2$ potência de 2.

Resolução de recorrências

Recorrência do Mergesort:

$$T(n) \; = \; T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$ e $T(n) = 2\,T(n/2) + n$ para $n \geq 2$ potência de 2.

Expansão:

$$\begin{aligned}
T(n) &= 2T(n/2) + n \\
&= 2\big(2T(n/2^2) + n/2\big) + n \; = \; 2^2T(n/2^2) + 2n \\
&= 2^2\big(2T(n/2^3) + n/2^2\big) + 2n \; = \; 2^3T(n/2^3) + 3n \\
&= 2^3\big(2T(n/2^4) + n/2^3\big) + 3n \; = \; 2^4T(n/2^4) + 4n \\
&= \ldots \; = \; 2^kT(n/2^k) + kn \quad \text{para } k = \lg n \\
&= n\,T(1) + (\lg n)n \; = \; n + n\lg n \; = \; \Theta(n\lg n)
\end{aligned}$$

Resolução de recorrências

$$T(n) \;=\; T(\lfloor n/2 \rfloor) + \Theta(1)$$

Resolução de recorrências

$$T(n) \;=\; T(\lfloor n/2 \rfloor) + \Theta(1)$$

Versão simplificada:

$T(1) = 1$ e $T(n) = T(n/2) + 1$ para $n \geq 2$ potência de 2.

Resolução de recorrências

$$T(n) \;=\; T(\lfloor n/2 \rfloor) + \Theta(1)$$

Versão simplificada:

$T(1) = 1$ e $T(n) = T(n/2) + 1$ para $n \geq 2$ potência de 2.

Expansão:

$$\begin{aligned}
T(n) &= T(n/2) + 1 \\
&= \big(T(n/2^2) + 1\big) + 1 \;=\; T(n/2^2) + 2 \\
&= \big(T(n/2^3) + 1\big) + 2 \;=\; T(n/2^3) + 3 \\
&= \big(T(n/2^4) + 1\big) + 3 \;=\; T(n/2^4) + 4 \\
&= \;\ldots\; = \; T(n/2^k) + k \quad \text{para } k = \lg n \\
&= T(1) + \lg n \;=\; 1 + \lg n \;=\; \Theta(\lg n)
\end{aligned}$$

Resolução de recorrências

$$T(n) \;=\; T(n-1) + \Theta(n)$$

Resolução de recorrências

$$T(n) \; = \; T(n-1) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$ e $T(n) \; = \; T(n-1) + n$ para $n \geq 2$.

Resolução de recorrências

$$T(n) \;=\; T(n-1) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$ e $T(n) \;=\; T(n-1) + n$ para $n \geq 2$.

Expansão:

$$\begin{aligned}
T(n) &= T(n-1) + n \\
&= T(n-2) + (n-1) + n \\
&= T(n-3) + (n-2) + (n-1) + n \\
&= T(n-4) + (n-3) + (n-2) + (n-1) + n \\
&= \ldots \;=\; T(1) + 2 + 3 + \cdots + (n-2) + (n-1) + n \\
&= 1 + 2 + \cdots + (n-1) + (n-2) + n \;=\; \frac{(n+1)n}{2} = \Theta(n^2)
\end{aligned}$$

Resolução de recorrências

Expansão:

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

Resolução de recorrências

Expansão:

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

Note que não temos restrição no n neste caso.

Só faça a conta quando
os termos que saem da recorrência são o mesmo
(como o n na recorrência do Mergesort, e o 1 na anterior).

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$

$T(n) = 3\,T(n/2) + n$ para $n \geq 2$ potência de 2.

$$T(n) \ = \ 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$

$T(n) = 3\,T(n/2) + n$ para $n \geq 2$ potência de 2.

Expansão:

$$\begin{aligned} T(n) &= \ 3T\Big(\frac{n}{2}\Big) + n \\ &= \ 3\Big(3T\Big(\frac{n}{2^2}\Big) + \frac{n}{2}\Big) + n \ = \ 3^2T\Big(\frac{n}{2^2}\Big) + \frac{3}{2}\,n + n \\ &= \ 3^2\Big(3T\Big(\frac{n}{2^3}\Big) + \frac{n}{2^2}\Big) + \frac{3}{2}\,n + n \ = \ 3^3T\Big(\frac{n}{2^3}\Big) + \Big(\frac{3}{2}\Big)^2 n + \frac{3}{2}\,n + n \end{aligned}$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

Versão simplificada:

$T(1) = 1$

$T(n) = 3\,T(n/2) + n$ para $n \geq 2$ potência de 2.

Expansão:

$$\begin{aligned}
T(n) &= 3T\big(\frac{n}{2}\big) + n \\
&= 3\Big(3T\big(\frac{n}{2^2}\big) + \frac{n}{2}\Big) + n \;=\; 3^2T\big(\frac{n}{2^2}\big) + \frac{3}{2}n + n \\
&= 3^2\Big(3T\big(\frac{n}{2^3}\big) + \frac{n}{2^2}\Big) + \frac{3}{2}n + n \;=\; 3^3T\big(\frac{n}{2^3}\big) + \big(\frac{3}{2}\big)^2 n + \frac{3}{2}n + n \\
&= 3^3\Big(3T\big(\frac{n}{2^4}\big) + \frac{n}{2^3}\Big) + \big(\frac{3}{2}\big)^2 n + \frac{3}{2}n + n \qquad (3) \\
&= 3^4T\big(\frac{n}{2^4}\big) + \big(\frac{3}{2}\big)^3 n + \big(\frac{3}{2}\big)^2 n + \frac{3}{2}n + n
\end{aligned}$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

Expansão:

$$\begin{aligned}
T(n) &= 3T(\frac{n}{2}) + n \;=\; 3^2T(\frac{n}{2^2}) + \frac{3}{2}\,n + n \\
&= 3^3T(\frac{n}{2^3}) + (\frac{3}{2})^2 n + \frac{3}{2}\,n + n \\
&= 3^4T(\frac{n}{2^4}) + (\frac{3}{2})^3 n + (\frac{3}{2})^2 n + \frac{3}{2}\,n + n \\
&= \;\ldots\; = 3^kT(\frac{n}{2^k}) + (\frac{3}{2})^{k-1} n + \cdots + \frac{3}{2}\,n + n \quad \text{para } k = \lg n
\end{aligned}$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

Expansão:

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

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

$$= 3^4T(\frac{n}{2^4}) + (\frac{3}{2})^3 n + (\frac{3}{2})^2 n + \frac{3}{2}\,n + n$$

$$= \ldots = 3^kT(\frac{n}{2^k}) + (\frac{3}{2})^{k-1} n + \cdots + \frac{3}{2}\,n + n \quad \text{para } k = \lg n$$

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

$$= 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1}\Big) n = 3^{\lg n} + 2\,\Big(\big(\frac{3}{2}\big)^{\lg n} - 1\Big) n$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$\begin{aligned}
T(n) &= 3T(\tfrac{n}{2}) + n \;=\; 3^2T(\tfrac{n}{2^2}) + \tfrac{3}{2}\,n + n \\
&= \;\ldots\; = \; 3^kT(\tfrac{n}{2^k}) + (\tfrac{3}{2})^{k-1}n + \cdots + \tfrac{3}{2}\,n + n \quad \text{para } k = \lg n \\
&= 3^{\lg n}\,T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2}\Big) n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1}\Big) n \\
&= 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1\Big) n \\
&= 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1\Big) n \\
&= 3^{\lg n} + 2\,\big(3^{\lg n} - n\big)
\end{aligned}$$

$$T(n) \;=\; 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$
\begin{aligned}
T(n) &= \; 3T(\tfrac{n}{2}) + n \;=\; 3^2 T(\tfrac{n}{2^2}) + \frac{3}{2}\,n + n \\
&= \; \ldots \;=\; 3^k T(\tfrac{n}{2^k}) + (\tfrac{3}{2})^{k-1} n + \cdots + \frac{3}{2}\,n + n \quad \text{para } k = \lg n \\
&= \; 3^{\lg n}\, T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2}\Big) n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1}\Big) n \\
&= \; 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1\Big) n \\
&= \; 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1\Big) n \\
&= \; 3^{\lg n} + 2\,\Big(3^{\lg n} - n\Big) \\
&= \; 3^{\lg n} + 2\,3^{\lg n} - 2n
\end{aligned}
$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$
\begin{aligned}
T(n) &= 3T(\tfrac{n}{2}) + n \;=\; 3^2T(\tfrac{n}{2^2}) + \tfrac{3}{2}\,n + n \\
&= \;\ldots\; = \; 3^kT(\tfrac{n}{2^k}) + (\tfrac{3}{2})^{k-1}n + \cdots + \tfrac{3}{2}\,n + n \quad \text{para } k = \lg n \\
&= 3^{\lg n}\,T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2}\Big)n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1}\Big)n \\
&= 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1\Big)n \;=\; 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1\Big)n \\
&= 3^{\lg n} + 2\,\big(3^{\lg n} - n\big) \\
&= 3^{\lg n} + 2\,3^{\lg n} - 2n \\
&= 3\,3^{\lg n} - 2n
\end{aligned}
$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

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

$$= 3^{\lg n} + 2\,\Big(\Big(\frac{3}{2}\Big)^{\lg n} - 1\Big)n = 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1\Big)n$$

$$= 3^{\lg n} + 2\,\big(3^{\lg n} - n\big)$$

$$= 3^{\lg n} + 2\,3^{\lg n} - 2n$$

$$= 3\,3^{\lg n} - 2n$$

$$T(n) \;=\; 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$
\begin{aligned}
T(n) &= \; 3^{\lg n}\,T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2}\Big) n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1}\Big) n \\
&= \; 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1\Big) n \;=\; 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1\Big) n \\
&= \; 3^{\lg n} + 2\,\big(3^{\lg n} - n\big) \\
&= \; 3^{\lg n} + 2\,3^{\lg n} - 2n \\
&= \; 3\,3^{\lg n} - 2n \\
&= \; 3\,\big(2^{\lg 3}\big)^{\lg n} - 2n
\end{aligned}
$$

$$T(n) \;=\; 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$\begin{aligned}
T(n) = \;& 3^{\lg n}\, T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2} \Big) n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1} \Big) n \\
=\;& 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1 \Big) n \;=\; 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1 \Big) n \\
=\;& 3^{\lg n} + 2\,\big(3^{\lg n} - n \big) \\
=\;& 3^{\lg n} + 2\, 3^{\lg n} - 2n \\
=\;& 3\, 3^{\lg n} - 2n \\
=\;& 3\, \big(2^{\lg n}\big)^{\lg 3} - 2n \\
=\;& 3\, n^{\lg 3} - 2n
\end{aligned}$$

$$T(n) = 3\,T(\lfloor n/2 \rfloor) + \Theta(n)$$

$$
\begin{aligned}
T(n) &= 3^{\lg n}\, T(1) + \Big(\sum_{i=0}^{\lg n - 1} \frac{3}{2} \Big) n \;=\; 3^{\lg n} + \Big(\frac{(\frac{3}{2})^{\lg n} - 1}{\frac{3}{2} - 1} \Big) n \\
&= 3^{\lg n} + 2\,\Big((\frac{3}{2})^{\lg n} - 1 \Big) n \;=\; 3^{\lg n} + 2\,\Big(\frac{3^{\lg n}}{n} - 1 \Big) n \\
&= 3^{\lg n} + 2\,\big(3^{\lg n} - n \big) \\
&= 3^{\lg n} + 2\,3^{\lg n} - 2n \\
&= 3\,3^{\lg n} - 2n \\
&= 3\,\big(2^{\lg n} \big)^{\lg 3} - 2n \\
&= 3\,n^{\lg 3} - 2n \\
&= \Theta(n^{\lg 3})
\end{aligned}
$$

Segmento de soma máxima

Um segmento de um vetor $A[1\,..\,n]$ é qualquer subvetor da forma $A[e\,..\,d]$.

Problema: Dado um vetor $A[1\,..\,n]$ de números inteiros, determinar um segmento $A[e\,..\,d]$ de soma máxima.

Entra:

$$
\begin{array}{c|c|c|c|c|c|c|c|c|c|c|}
 & 1 & & & & & & & & & n \\
\cline{2-11}
A & 31 & -41 & 59 & 26 & -53 & 58 & 97 & -93 & -23 & 84 \\
\cline{2-11}
\end{array}
$$

Segmento de soma máxima

Um segmento de um vetor $A[1\,..\,n]$ é qualquer subvetor da forma $A[e\,..\,d]$.

Problema: Dado um vetor $A[1\,..\,n]$ de números inteiros, determinar um segmento $A[e\,..\,d]$ de soma máxima.

Entra:

	1									n
A	31	−41	59	26	−53	58	97	−93	−23	84

Sai:

	1		3				7			n
A	31	−41	59	26	−53	58	97	−93	−23	84

$A[e\,..\,d] = A[3\,..\,7]$ é segmento de soma máxima.
$A[3\,..\,7]$ tem soma 187.

Algoritmo café-com-leite

Determina um segmento de soma máxima de $A[1\,.\,.\,n]$.

SEG-MAX-3 (A, n)

$somamax \leftarrow 0$
$e \leftarrow 0 \quad d \leftarrow -1 \quad \triangleright\ A[e\,.\,.\,d]$ é vazio
para $i \leftarrow 1$ **até** n **faça**
    para $f \leftarrow i$ **até** n **faça**
        $soma \leftarrow 0$
        para $k \leftarrow i$ **até** f **faça**
            $soma \leftarrow soma + A[k]$
        se $soma > somamax$ **então**
            $somamax \leftarrow soma \quad e \leftarrow i \quad d \leftarrow f$
devolva e, d **e** $somamax$

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é:

linha		todas as execuções da linha	
1-2	$=$	2	$= \Theta(1)$
3	$=$	$n+1$	$= \Theta(n)$
4	$=$	$(n+1)+n+(n-1)+\cdots+2$	$= \Theta(n^2)$
5	$=$	$n+(n-1)+\cdots+1$	$= \Theta(n^2)$
6	$=$	$(2+\cdots+(n+1))+(2+\cdots+n)+\cdots+2$	$= \Theta(n^3)$
7	$=$	$(1+\cdots+n)+(1+\cdots+(n-1))+\cdots+1$	$= \Theta(n^3)$
8	$=$	$n+(n-1)+(n-2)+\cdots+1$	$= \Theta(n^2)$
9	$\leq$	$n+(n-1)+(n-2)+\cdots+1$	$= \mathrm{O}(n^2)$
10	$=$	1	$= \Theta(1)$
total	$=$	$\Theta(2n^3+3n^2+n+2)+\mathrm{O}(n^2)$	$= \Theta(n^3)$

Algoritmo arroz-com-feijão

Determina um segmento de soma máxima de $A[1\,..\,n]$.

SEG-MAX-2 (A, n)

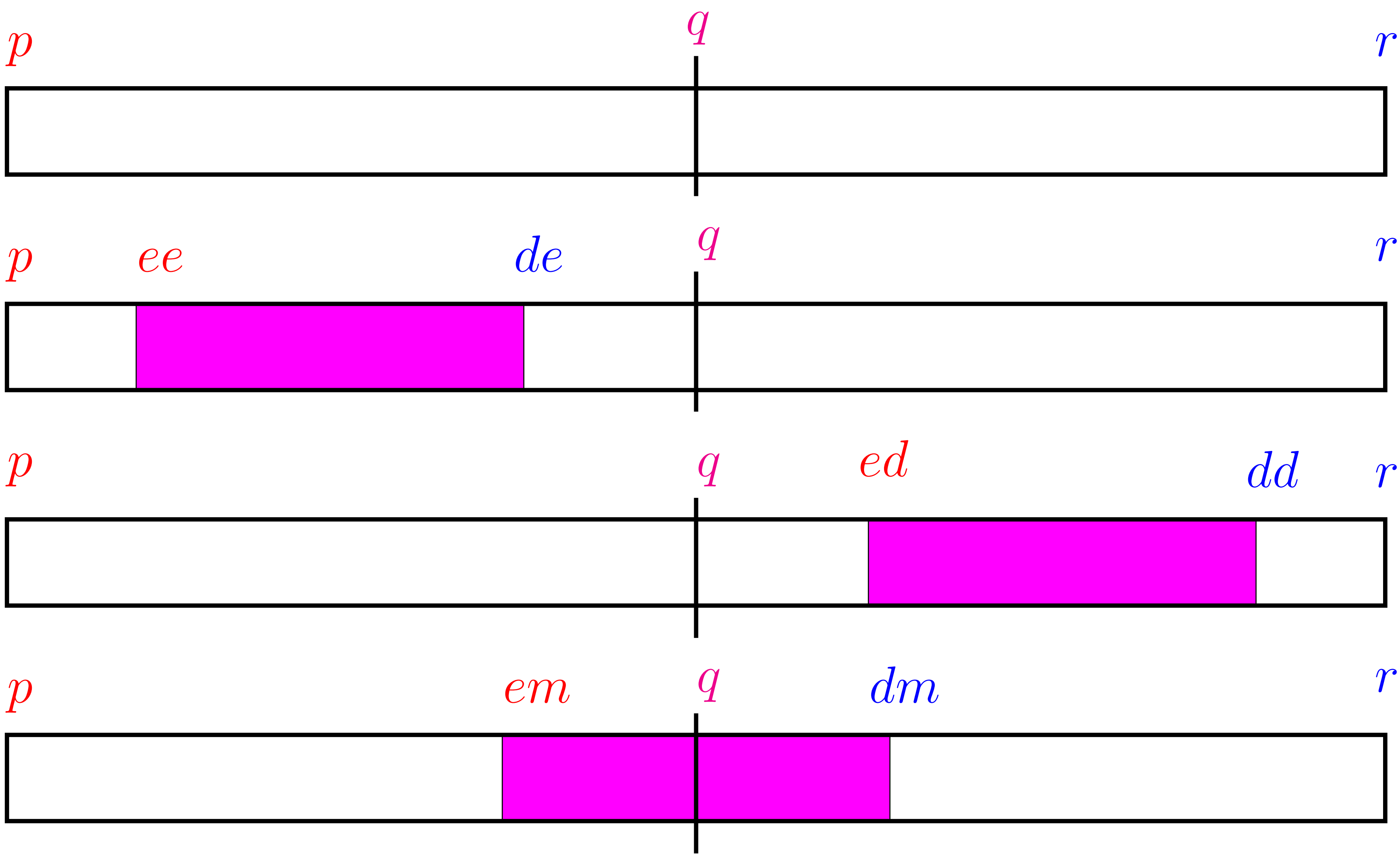
$somamax \leftarrow 0$
$e \leftarrow 0 \quad d \leftarrow -1 \quad \triangleright\ A[e\,..\,d]$ é vazio
para $i \leftarrow 1$ **até** n **faça**
    $soma \leftarrow 0$
    para $f \leftarrow i$ **até** n **faça**
        $soma \leftarrow soma + A[f]$
        se $soma > somamax$ **então**
            $somamax \leftarrow soma \quad e \leftarrow i \quad d \leftarrow f$
devolva e, d **e** $somamax$

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é:

linha		todas as execuções da linha	
1-2	$=$	2	$= \Theta(1)$
3	$=$	$n+1$	$= \Theta(n)$
4	$=$	n	$= \Theta(n)$
5	$=$	$(n+1)+n+\cdots+2$	$= \Theta(n^2)$
6	$=$	$n+(n-1)+\cdots+1$	$= \Theta(n^2)$
7	$=$	$n+(n-1)+\cdots+1$	$= \Theta(n^2)$
8	$\leq$	$n+(n-1)+\cdots+1$	$= \mathrm{O}(n^2)$
9	$=$	1	$= \Theta(1)$
total	$=$	$\Theta(3n^2+2n+2)+\mathrm{O}(n^2)$	$= \Theta(n^2)$

Solução de divisão-e-conquista



Algoritmo de divisão-e-conquista

Determina soma máxima de um seg. de $A[p\,..\,r]$.

SEG-MAX-DC (A, p, r)

se $p = r$ **então devolva** $\max(0, A[p])$
$q \leftarrow \lfloor (p + r)/2 \rfloor$
$maxesq \leftarrow$ SEG-MAX-DC(A, p, q)
$maxdir \leftarrow$ SEG-MAX-DC$(A, q+1, r)$
$max2esq \leftarrow soma \leftarrow A[q]$
para $i \leftarrow q - 1$ **decrescendo até** p **faça**
    $soma \leftarrow soma + A[i]$
    $max2esq \leftarrow \max(max2esq, soma)$
$max2dir \leftarrow soma \leftarrow A[q+1]$
para $f \leftarrow q + 2$ **até** r **faça**
    $soma \leftarrow soma + A[f]$
    $max2dir \leftarrow \max(max2dir, soma)$
$maxcruz \leftarrow max2esq + max2dir$
devolva $\max(maxesq, maxcruz, maxdir)$

Correção

Verifique que:

- $maxesq$ é a soma máxima de um segmento de $A[p\,..\,q]$;
- $maxdir$ é a soma máxima de um segmento de $A[q+1\,..\,r]$; e
- $maxcruz$ é a soma máxima de um segmento da forma $A[i\,..\,f]$ com $i \leq q$ e $q+1 \leq f$.

Conclua que o algoritmo devolve a soma máxima de um segmento de $A[p\,..\,r]$.

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é:

linha		todas as execuções da linha	
1-2	$=$	2	$= \Theta(1)$
3	$=$	$T(\lceil \frac{n}{2} \rceil)$	$= T(\lceil \frac{n}{2} \rceil)$
4	$=$	$T(\lfloor \frac{n}{2} \rfloor)$	$= T(\lfloor \frac{n}{2} \rfloor)$
5	$=$	1	$= \Theta(1)$
6	$=$	$\lceil \frac{n}{2} \rceil + 1$	$= \Theta(n)$
7-8	$=$	$\lceil \frac{n}{2} \rceil$	$= \Theta(n)$
9	$=$	1	$= \Theta(1)$
10	$=$	$\lfloor \frac{n}{2} \rfloor + 1$	$= \Theta(n)$
11-12	$=$	$\lfloor \frac{n}{2} \rfloor$	$= \Theta(n)$
13-14	$=$	2	$= \Theta(1)$
total	$=$	$T(\lceil \frac{n}{2} \rceil) + T(\lfloor \frac{n}{2} \rfloor) + \Theta(n)$	

Consumo de tempo

$T(n) :=$ consumo de tempo quando $n = r - p + 1$

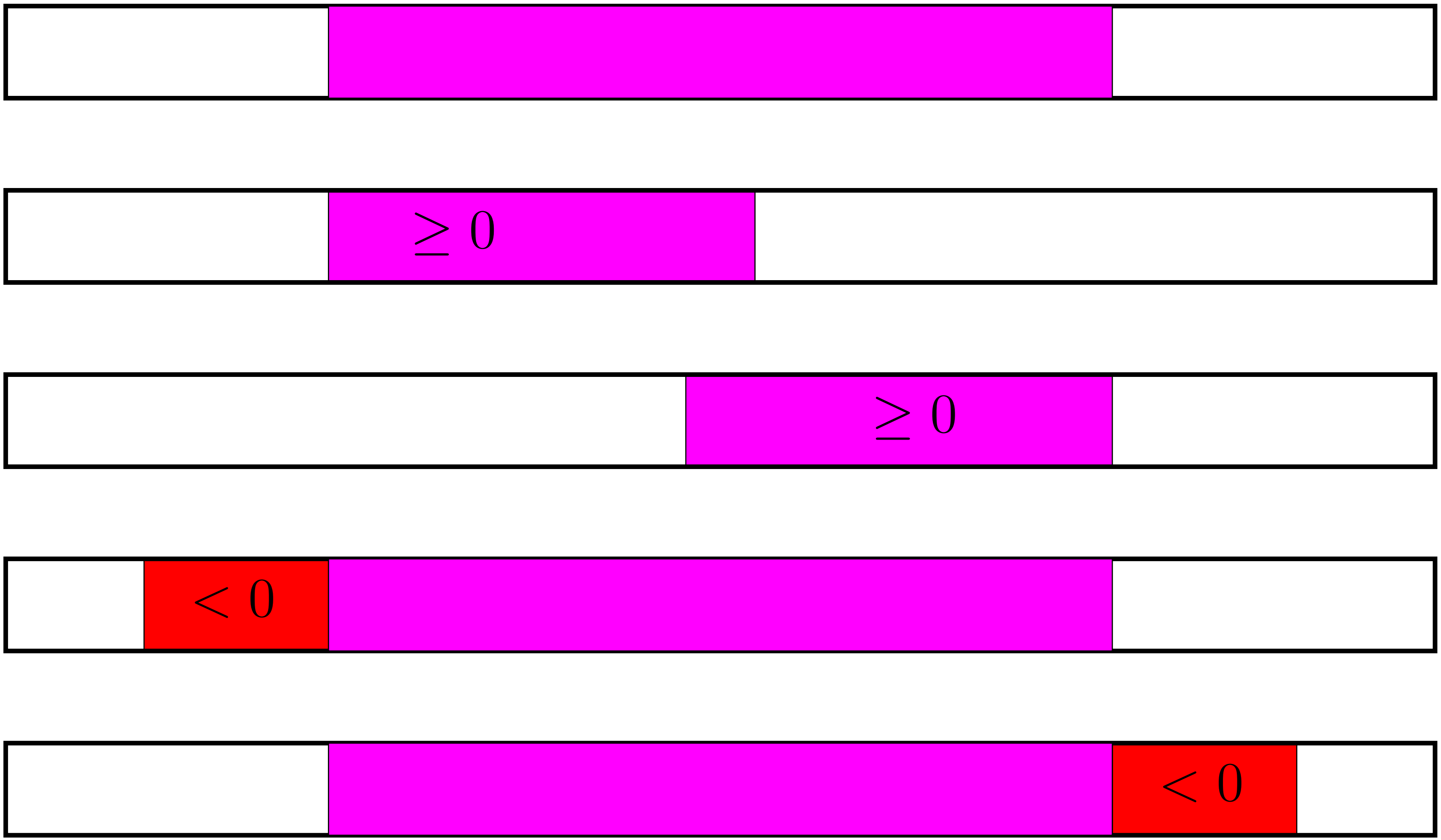
Na análise do consumo de tempo do SEG-MAX-DC chegamos a (já manjada) recorrência com Θ do lado direito:

$$T(n) = T(\lceil n/2 \rceil) + T(\lfloor n/2 \rfloor) + \Theta(n)$$

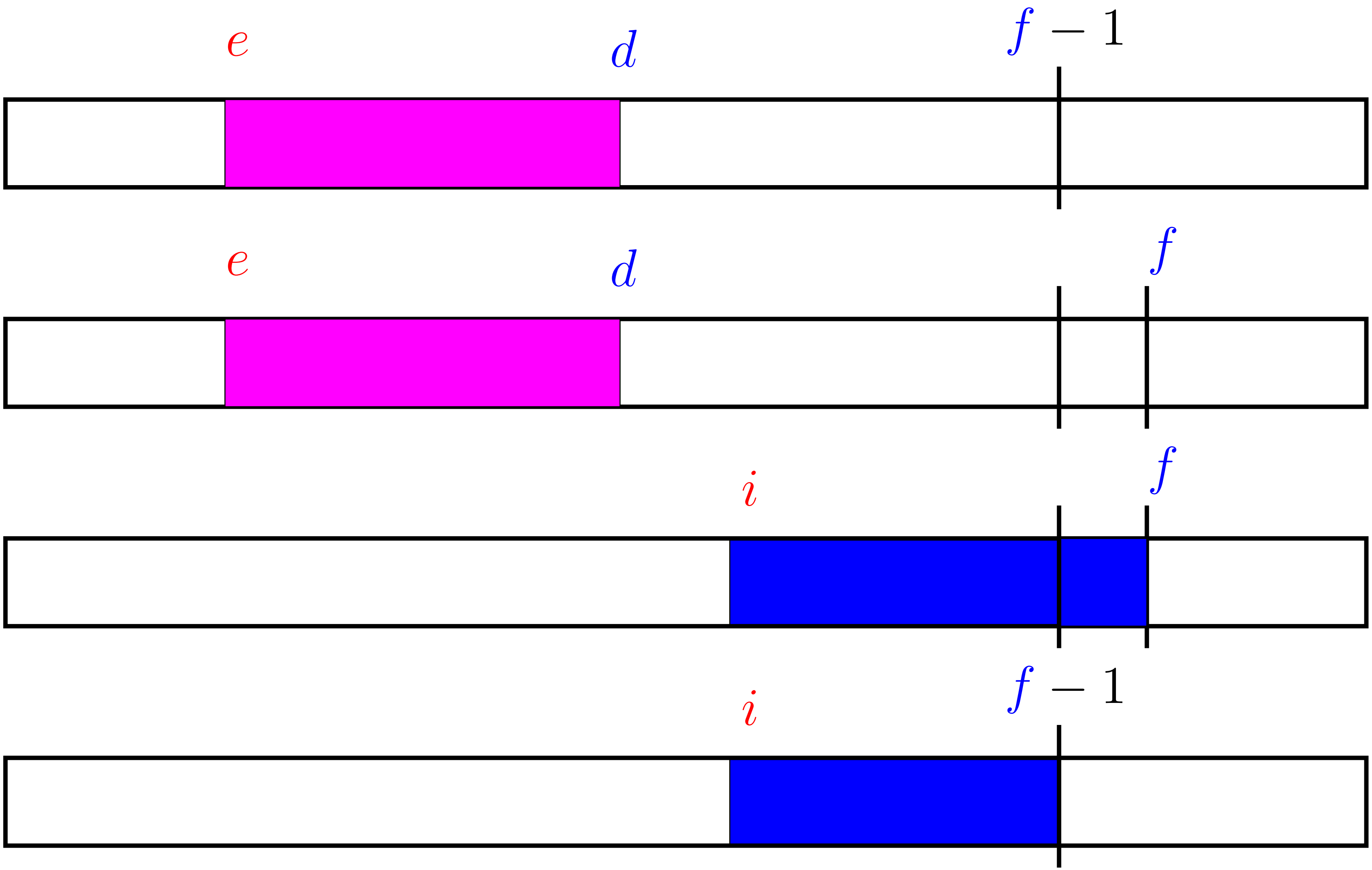
Solução assintótica: $T(n)$ é $\Theta(n \lg n)$.

Cara da solução

Solução



Solução indutiva



Algoritmo linear

Determina um segmento de soma máxima de $A[1\,.\,.\,n]$ (por Jay Kadane).

SEG-MAX-1 (A, n)

$somamax \leftarrow 0$
$e \leftarrow 0 \quad d \leftarrow -1 \quad \triangleright\ A[e\,.\,.\,d]$ é vazio
$i \leftarrow 1$
$soma \leftarrow 0$
para $f \leftarrow 1$ **até** n **faça**
    se $soma + A[f] < 0$
        então $i \leftarrow f + 1 \quad soma \leftarrow 0$
        senão $soma \leftarrow soma + A[f]$
    se $soma > somamax$ **então**
        $somamax \leftarrow soma \quad e \leftarrow i \quad d \leftarrow f$
devolva e, d **e** $somamax$

Correção

Verifique que:

- $A[e \,..\, d]$ é um segmento de soma máxima em $A[1 \,..\, f-1]$.
- $somamax$ é a soma de $A[e \,..\, d]$.
- $A[i \,..\, f-1]$ é um segmento de soma máxima que termina em $f-1$.
- $soma$ é a soma de $A[i \,..\, f-1]$.

Conclua que o algoritmo devolve a soma máxima de um segmento de $A[1 \,..\, n]$.

Consumo de tempo

Se a execução de cada linha de código consome 1 unidade de tempo o consumo total é:

linha		todas as execuções da linha	
1-2	$=$	2	$= \Theta(1)$
3-4	$=$	2	$= \Theta(1)$
5	$=$	$n+1$	$= \Theta(n)$
6	$=$	n	$= \Theta(n)$
7-8	$=$	n	$= \Theta(n)$
9	$=$	n	$= \Theta(n)$
10	$\leq$	n	$= \mathrm{O}(n)$
11	$=$	1	$= \Theta(1)$
total	$=$	$\Theta(4n+3) + \mathrm{O}(n)$	$= \Theta(n)$

Conclusões

O consumo de tempo do algoritmo SEG-MAX-3 é $\Theta(n^3)$.

O consumo de tempo do algoritmo SEG-MAX-2 é $\Theta(n^2)$.

O consumo de tempo do algoritmo SEG-MAX-DC é $\Theta(n \lg n)$.

O consumo de tempo do algoritmo SEG-MAX-1 é $\Theta(n)$.

Técnicas

- Evitar recálculos. Usar espaço para armazenar resultados a fim de evitar recalculá-los (SEG-MAX-2, SEG-MAX-1, programação dinâmica).
- Divisão-e-conquista. Os algoritmos Mergesort e SEG-MAX-2 utilizam uma forma conhecida dessa técnica.
- Algoritmos incrementais/varredura. Como estender a solução de um subproblema a uma solução do problema (SEG-MAX-1).
- Delimitação inferior. Bons projetistas de algoritmos só dormem em paz quando sabem que seus algoritmos são o melhor possível (SEG-MAX-1).