Aula 10 -Algoritmo de Seleção

Seleção Linear

Prof. Marco Aurélio Stefanes
`marco em dct.ufms.br`
`www.dct.ufms.br/ ~ marco`

`Baseado no slide do prof. Fiofilof`

i-ésimo menor elemento

Problema: Encontrar o i-ésimo menor elemento de $A[1,\ldots,n]$

Suponha $A[1\,..\,n]$ sem elementos repetidos.

Exemplo: 33 é o 4o. menor elemento de:

1									10	
22	99	32	88	34	**33**	11	97	55	66	A

1			4						10	
11	22	32	**33**	34	55	66	88	97	99	ordenado

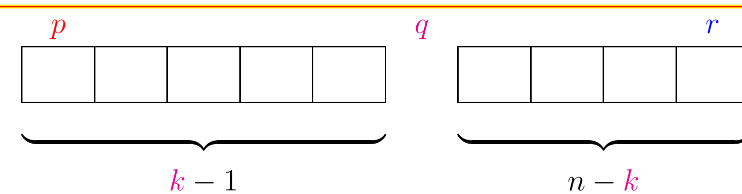
SelectB

Recebe $A[p\,..\,r]$ e i tal que $1 \le i \le r-p+1$ e devolve um índice q tal que $A[q]$ é o i-ésimo menor elemento de $A[p\,..\,r]$

SELECT-B(A, p, r, i)

```
if p = r then
    devolva p        ▷ p e não A[p]
q ← PARTICIONE-B (A, p, r)
k ← q − p + 1
if k = i then
    devolva q        ▷ q e não A[q]
if k > i then
    devolva SELECT-B (A, p, q − 1, i)
else
    devolva SELECT-B (A, q + 1, r, i − k)
```


Particione-B



Rearranja $A[p\,..\,r]$ e devolve um índice q, $p \le q \le r$, tal que $A[p\,..\,q-1] \le A[q] < A[q+1\,..\,r]$ **e**

$$\max\{k-1, n-k\} \le \left\lfloor \frac{7n}{10} \right\rfloor + 6,$$

onde $n = r - p + 1$ e $k = q - p + 1$.

Suponha que

$P(n) :=$ consumo de tempo máximo do algoritmo PARTICIONE-B quando $n = r - p + 1$

Consumo de tempo

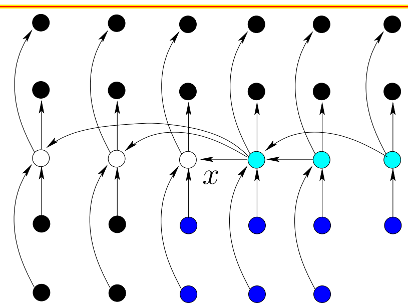
$T(n) :=$ consumo de tempo **máximo** do algoritmo SELECT-B quando $n = r - p + 1$

linha	consumo de todas as execuções da linha
1-2	$= 2\,\Theta(1)$
3	$= P(n)$
4-7	$= 4\,\Theta(1)$
8	$= T(k-1)$
10	$= T(n-k)$

$$T(n) = 6\,\Theta(1) + P(n) + \max\{T(k-1), T(n-k)\}$$

$$\leq \Theta(1) + P(n) + T(\lfloor \tfrac{7n}{10} \rfloor + 6)$$

Particione-B



$$\max\{k-1, n-k\} \leq n - 3\left(\left\lceil \frac{1}{2}\left\lceil \frac{n}{5} \right\rceil \right\rceil - 2\right)$$

$$\leq n - \left(\frac{3n}{10} - 6\right) = \frac{7n}{10} + 6$$

Particione-B

$n := r - p + 1$

ParticioneB (A, p, r)

1: **for** $j \leftarrow p, p{+}5, p{+}5\cdot 2, \ldots$ to $p{+}5(\lceil n/5 \rceil{-}1)$ **do**
2:     ORDENE $(A, j, j{+}4)$
3: ORDENE $(A, p{+}5\lfloor n/5 \rfloor, n)$
4: **for** $j \leftarrow 1$ to $\lceil n/5 \rceil{-}1$ **do**
5:     $B[j] \leftarrow A[p{+}5j{-}3]$
6: $B[\lceil n/5 \rceil] \leftarrow A[\lfloor (p{+}5\lfloor n/5 \rfloor{+}n)/2 \rfloor]$
7: $k \leftarrow$ SELECT-B$(B, 1, \lceil n/5 \rceil, \lfloor (\lceil n/5 \rceil{+}1)/2 \rfloor)$
8: $A[k] \leftrightarrow A[r]$
9: devolva PARTICIONE (A, p, r)

Complexidade do Particione-B

$P(n) :=$ consumo de tempo **máximo** do algoritmo PARTICIONE-B quando $n = r - p + 1$

linha	consumo de todas as execuções da linha
1-3	$= \lceil n/5 \rceil\,\Theta(1)$
4-6	$= \lceil n/5 \rceil\,\Theta(1)$
7	$= T(\lceil n/5 \rceil)$
8	$= \Theta(1)$
9	$= \Theta(n)$

$$P(n) = \Theta(2\lceil n/5 \rceil + n + 1) + T(\lceil n/5 \rceil)$$

$$= \Theta(n) + T(\lceil n/5 \rceil)$$

Complexidade do Select-B

$T(n) :=$ consumo de tempo máximo do algoritmo
SELECT-B quando $n = r - p + 1$

Temos que

$$T(1) = \Theta(1)$$

$$\begin{aligned} T(n) &\leq \Theta(1) + P(n) + T\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) \\ &\leq \Theta(1) + \Theta(n) + T\left(\left\lceil \frac{n}{5} \right\rceil\right) + T\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) \\ &= \Theta(n) + T\left(\left\lceil \frac{n}{5} \right\rceil\right) + T\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) \end{aligned}$$

para $n = 2, 3, \ldots$,

Complexidade do Select-B

$T(n)$ pertence a mesma classe O que:

$$S(n) = 1 \quad \text{para } n < 30$$

$$S(n) \leq S\left(\left\lceil \frac{n}{5} \right\rceil\right) + S\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) + n \quad \text{para } n \geq 30$$

n	30	60	90	120	150	180	210	240	270	300
$S(n)$	32	185	330	451	572	732	902	1040	1224	1439

Vamos verificar que $S(n) < 80\,n$ para $n = 1, 2, 3, 4, \ldots$

prova: Se $n = 1, \ldots, 29$, então $S(n) = 1 < 80 < 80n$.
Se $n = 30, \ldots, 99$, então

$$S(n) < S(120) = 451 < 80 \times 30 \leq 80n.$$

Recorrência

Se $n \geq 100$, então

$$\begin{aligned} S(n) &\leq S\left(\left\lceil \frac{n}{5} \right\rceil\right) + S\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) + n \\ &\stackrel{\text{hi}}{<} 80\left\lceil \frac{n}{5} \right\rceil + 80\left(\left\lfloor \frac{7n}{10} \right\rfloor + 6\right) + n \\ &\leq 80\left(\frac{n}{5} + 1\right) + 80\left(\frac{7n}{10} + 6\right) + n \\ &= 80\frac{n}{5} + 80 + 80\frac{7n}{10} + 480 + n \\ &= 16n + 56n + n + 560 \\ &= 73n + 560 \\ &< 80n \quad \text{(pois } n \geq 100\text{).} \end{aligned}$$

Logo, $T(n)$ é $\mathrm{O}(n)$.

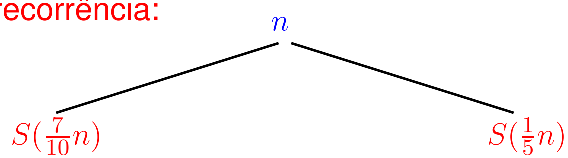
Como adivinhar a classe O?

Árvore da recorrência:

$$S(n)$$

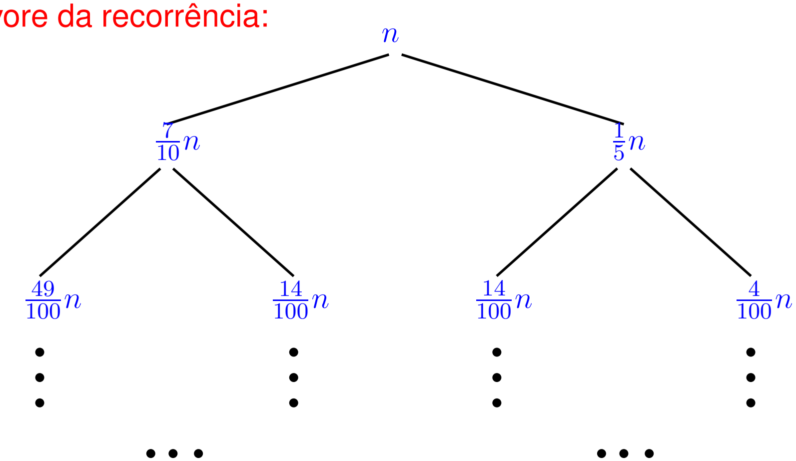
Como adivinhar a classe O?

Árvore da recorrência:



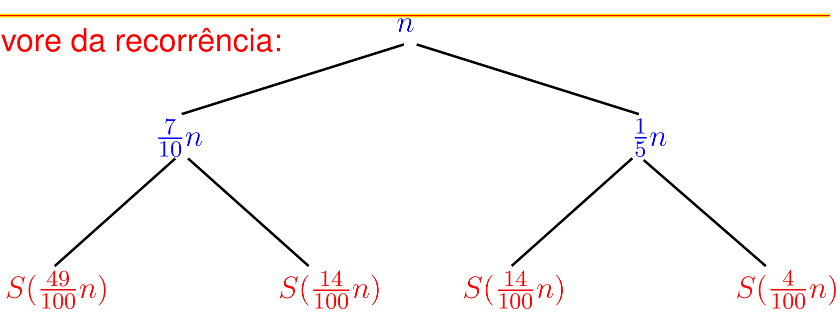
Como adivinhar a classe O?

Árvore da recorrência:



Como adivinhar a classe O?

Árvore da recorrência:



Contas

nível	0	1	2	$\ldots$	$k-1$	k
soma	n	$\frac{9}{10}n$	$\frac{9^2}{10^2}n$	$\ldots$	$\frac{9^{k-1}}{10^{k-1}}n$	$\frac{9^k}{10^k}n$

$$\frac{10^{k-1}}{9^{k-1}} < n \leq \frac{10^k}{9^k} \Rightarrow k = \lceil \log_{\frac{10}{9}} n \rceil$$

$$\begin{aligned} S(n) &= n + \frac{9}{10}n + \cdots + \frac{9^{k-1}}{10^{k-1}}n + \frac{9^k}{10^k}n \\ &= (1 + \frac{9}{10} + \cdots + \frac{9^k}{10^k})n \\ &= 10(1 - \frac{9^{k+1}}{10^{k+1}})n \\ &< 10n \end{aligned}$$

Conclusão

O consumo de tempo do algoritmo SELECT-B no pior caso é $O(n)$.

O algoritmo roda em tempo linear se dividirmos os elementos em grupos de 3? e em grupos de 7? Porque?