AULA 11

Quicksort

CLRS 7

Quicksort

Rearranja $A[p \mathinner{..} r]$ em ordem crescente.

QUICKSORT (A, p, r)
1 se $p < r$
2 então $q \leftarrow$ PARTICIONE (A, p, r)
3 QUICKSORT $(A, p, q-1)$
4 QUICKSORT $(A, q+1, r)$

p					q					r
22	22	22	11	11	44	88	99	88	99	88

No começo da linha 3,

$$A[p \mathinner{..} q{-}1] \leq A[q] \leq A[q{+}1 \mathinner{..} r]$$

PARTICIONE rearranja A de modo que
$p \leq q \leq r$ e $A[p\,.\,.\,q-1] \leq A[q] \leq A[q+1\,.\,.\,r]$

PARTICIONE (A, p, r)
1 $x \leftarrow A[r]$ $\quad\triangleright$ x é o “pivô”
2 $i \leftarrow p - 1$
3 para $j \leftarrow p$ até $r - 1$
4 faça se $A[j] \leq x$
5 então $i \leftarrow i + 1$
6 $A[i] \leftrightarrow A[j]$
7 $A[i+1] \leftrightarrow A[r]$
8 devolva $i + 1$

Invariantes: no começo de cada iteração de 3–6,
$A[p\,.\,.\,i] \leq x \qquad A[i+1\,.\,.\,j-1] > x \qquad A[r] = x$

p		i				j				r
22	22	22	88	88	88	11	99	11	99	44

Consumo de tempo de PARTICIONE é $\Theta(n)$,
sendo $n = r - p + 1$.

Consumo de tempo do Quicksort

```
QUICKSORT (A, p, r)
se p < r
    então q ← PARTICIONE (A, p, r)
          QUICKSORT (A, p, q − 1)
          QUICKSORT (A, q + 1, r)
```

$T(n)$:= consumo de tempo no pior caso sendo $n := r - p + 1$

Recorrência grosseira:

$T(n) = T(0) + T(n-1) + \Theta(n)$

Solução: $T(n) = \Theta(n^2)$

Recorrência cuidadosa:

$$T(n) \;=\; \max_{0 \le k \le n-1} \left\{ T(k) + T(n-k-1) \right\} + \Theta(n)$$

Solução: $T(n) = \Theta(n^2)$

Prova?

EXEMPLO CONCRETO: $S(0) = S(1) = 1$ e

$$S(n) = \max_{0 \le k \le n-1} \Big\{ S(k) + S(n-k-1) \Big\} + n$$

para $n \ge 2$.

n	$S(n)$
0	1
1	1
2	$2+2$
3	$5+3$
4	$9+4$

Vou provar que $S(n) \le n^2 + 1$ para $n \ge 0$.

Prova: Trivial para $n \le 1$. Se $n \ge 2$ então

$$\begin{aligned} S(n) &= \max_{0 \le k \le n-1} \Big\{ S(k) + S(n-k-1) \Big\} + n \\ &\overset{\text{hi}}{\le} \max_{0 \le k \le n-1} \Big\{ k^2 + 1 + (n-k-1)^2 + 1 \Big\} + n \\ &= (n-1)^2 + 2 + n \qquad \rhd \text{ exerc 11.E} \\ &= n^2 - n + 3 \\ &\le n^2 + 1\,. \end{aligned}$$

Prove que $S(n) \ge \frac{1}{2} n^2$ para $n \ge 1$.

Tempo do QUICKSORT no melhor caso:

$$R(n) = R(\left\lfloor\frac{n-1}{2}\right\rfloor) + R(\left\lceil\frac{n-1}{2}\right\rceil) + \Theta(n)$$

Solução: $R(n) = \Theta(n \lg n)$

TAREFA 11

Exercício 11.A
Submeta ao algoritmo PARTICIONE um vetor com n elementos iguais. Como o algoritmo permuta o vetor recebido? Quantas trocas faz (linhas 6 e 7) entre elementos do vetor?

Exercício 11.B [CLRS 7.2-2]
Qual o consumo de tempo do QUICKSORT quando aplicado a um vetor com n elementos iguais?

Exercício 11.C [CLRS 7.2-3]
Mostre que o consumo de tempo do QUICKSORT é $\Omega(n^2)$ quando aplicado a um vetor crescente com n elementos distintos.

Exercício 11.D [CLRS 7.4-1, modificado]
Seja S a função definida sobre os inteiros positivos pela seguinte recorrência: $S(0) = S(1) = 1$ e

$$S(n) = \max_{0 \le k \le n-1} \{S(k) + S(n-k-1)\} + n$$

quando $n \geq 2$. Mostre que $S(n) \geq \frac{1}{2}n^2$ para $n \geq 1$.

Exercício 11.E [CLRS 7.4-3]
Mostre que $k^2 + (n-k-1)^2$ atinge o máximo para $0 \leq k \leq n-1$ quando $k = 0$ ou $k = n-1$.

Exercício 11.F
É verdade que $\lceil 2\lceil 2n/3\rceil/3\rceil = \lceil 4n/9\rceil$? É verdade que $\lfloor 2\lfloor 2n/3\rfloor/3\rfloor = \lfloor 4n/9\rfloor$?

Exercício 11.G [CLRS 7-4]
Considere a seguinte variante do algoritmo Quicksort:

```
QUICKSORT' (A, p, r)
  enquanto p < r faça
      q ← PARTICIONE (A, p, r)
      QUICKSORT' (A, p, q − 1)
      p ← q + 1
```

Mostre que a pilha de recursão pode atingir altura proporcional a n, onde $n := r - p + 1$. Modifique o código de modo que a pilha de recursão tenha altura $O(\lg n)$. (Veja enunciado completo em CLRS p.162.)