Análise de Algoritmos

Slides de Paulo Feofiloff

[com erros do coelho e agora também da cris]

Quicksort

CLRS 7

Partição

Problema: Rearranjar um dado vetor $A[p\,..\,d]$ e devolver um índice q tal que $p \le q \le d$ e

$$A[p\,..\,q-1] \le A[q] < A[q+1\,..\,d]$$

Entra:

	p									d
A	99	33	55	77	11	22	88	66	33	44

Problema: Rearranjar um dado vetor $A[p\,..\,d]$ e devolver um índice q tal que $p \le q \le d$ e

$$A[p\,..\,q-1] \le A[q] < A[q+1\,..\,d]$$

Entra:

	p									d
A	99	33	55	77	11	22	88	66	33	44

Sai:

	p				q					d
A	33	11	22	33	44	55	99	66	77	88

Particione

	p									d
A	99	33	55	77	11	22	88	66	33	44

Particione

i	j									x
A	99	33	55	77	11	22	88	66	33	44

i		j								x
A	99	33	55	77	11	22	88	66	33	44

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
	i				j					x
A	33	99	55	77	11	22	88	66	33	44

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
	i			j						x
A	33	99	55	77	11	22	88	66	33	44

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44
			i				j			x
A	33	11	22	77	99	55	88	66	33	44

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44
			i				j			x
A	33	11	22	77	99	55	88	66	33	44
			i					j		x
A	33	11	22	77	99	55	88	66	33	44

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44
			i				j			x
A	33	11	22	77	99	55	88	66	33	44
				i						j
A	33	11	22	33	99	55	88	66	77	44
	p				q					d
A	33	11	22	33	44	55	88	66	77	99

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44
			i				j			x
A	33	11	22	77	99	55	88	66	33	44
			i						j	x
A	33	11	22	77	99	55	88	66	33	44

Particione

Rearranja $A[p\,.\,.\,d]$ de modo que $p \le q \le d$ e
$A[p\,.\,.\,q{-}1] \le A[q] < A[q{+}1\,.\,.\,d]$

PARTICIONE (A, p, d)
1 $x \leftarrow A[d]$ $\triangleright$ x é o "pivô"
2 $i \leftarrow p{-}1$
3 **para** $j \leftarrow p$ **até** $d-1$ **faça**
4 **se** $A[j] \le x$
5 **então** $i \leftarrow i+1$
6 $A[i] \leftrightarrow A[j]$
7 $A[i{+}1] \leftrightarrow A[d]$
8 **devolva** $i+1$

Invariantes: no começo de cada iteração de 3–6,
(i0) $A[p\,.\,.\,i] \le x$ (i1) $A[i{+}1\,.\,.\,j{-}1] > x$ (i2) $A[d] = x$

Particione

i		j								x
A	99	33	55	77	11	22	88	66	33	44
	i		j							x
A	33	99	55	77	11	22	88	66	33	44
		i				j				x
A	33	11	55	77	99	22	88	66	33	44
			i				j			x
A	33	11	22	77	99	55	88	66	33	44
				i						j
A	33	11	22	33	99	55	88	66	77	44

Quicksort

Rearranja $A[p\,.\,.\,d]$ em ordem crescente.

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

	p									d
A	99	33	55	77	11	22	88	66	33	44

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

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

	p				q					d
A	33	11	22	33	44	55	88	66	77	99

No começo da linha 3,

$$A[p \mathinner{.\,.} q-1] \le A[q] \le A[q+1 \mathinner{.\,.} d]$$

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

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

No começo da linha 3,

$$A[p \mathinner{.\,.} q-1] \le A[q] \le A[q+1 \mathinner{.\,.} d]$$

Consumo de tempo?

Quicksort

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

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

	p				q					d
A	11	22	33	33	44	55	88	66	77	99

Quicksort

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

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

No começo da linha 3,

$$A[p \mathinner{.\,.} q-1] \le A[q] \le A[q+1 \mathinner{.\,.} d]$$

Consumo de tempo?

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

Quicksort

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

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

	p				q					d
A	11	22	33	33	44	55	66	77	88	99

Consumo de tempo

Quanto tempo consome em função de $n := d - p + 1$?

linha		consumo de todas as execuções da linha
1	=	?
2	=	?
3	=	?
4	=	?
total	=	????

Quanto tempo consome em função de $n := d - p + 1$?

linha		consumo de todas as execuções da linha
1	$=$	$\Theta(1)$
2	$=$	$\Theta(n)$
3	$=$	$T(k)$
4	$=$	$T(n-k-1)$
total	$=$	$T(k) + T(n-k-1) + \Theta(n+1)$

$$0 \le k := q - p \le n - 1$$

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

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

Recorrência grosseira:

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

$T(n)$ é $\Theta(n^2)$.

Demonstração: ... Exercício!

Recorrência

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

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

Recorrência cuidadosa

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

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

Recorrência

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

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

Recorrência grosseira:

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

$T(n)$ é $\Theta(???)$.

Versão simplificada

$T(0) = 1$

$T(1) = 1$

$T(n) = \max_{0 \le k \le n-1} \{T(k) + T(n-k-1)\} + n$ para $n = 2, 3, 4, \ldots$

n	0	1	2	3	4	5
$T(n)$	1	1	$2+2$	$5+3$	$9+4$	$14+5$

$T(0) = 1$

$T(1) = 1$

$T(n) = \max_{0 \le k \le n-1} \{T(k) + T(n-k-1)\} + n$ para $n = 2, 3, 4, \ldots$

n	0	1	2	3	4	5
$T(n)$	1	1	$2+2$	$5+3$	$9+4$	$14+5$

Vamos mostrar que $T(n) \le n^2 + 1$ para $n \ge 0$.

O consumo de tempo do QUICKSORT no caso médio é ???.

Partição $\frac{1}{10}$ para $\frac{9}{10}$:

$$R(n) = R(\lfloor \tfrac{n-1}{10} \rfloor) + R(\lceil \tfrac{9n-9}{10} \rceil) + \Theta(n)$$

Solução: $R(n)$ é $\Theta(n \lg n)$

Demonstração: ... Exercício!

O caso médio

O consumo de tempo do QUICKSORT no caso médio é ???.

Partição $\frac{1}{10}$ para $\frac{9}{10}$:

$$R(n) = R(\lfloor \tfrac{n-1}{10} \rfloor) + R(\lceil \tfrac{9n-9}{10} \rceil) + \Theta(n)$$

O caso médio

O consumo de tempo do QUICKSORT no caso médio é ???.

Partição $\frac{1}{10}$ para $\frac{9}{10}$:

$$R(n) = R(\lfloor \tfrac{n-1}{10} \rfloor) + R(\lceil \tfrac{9n-9}{10} \rceil) + \Theta(n)$$

Solução: $R(n)$ é $\Theta(n \lg n)$

Demonstração: ... Exercício!

Isso sugere que consumo médio é $\Theta(n \lg n)$.

Confirmação?

O caso médio

O consumo de tempo do QUICKSORT no caso médio é ???.

Partição $\frac{1}{10}$ para $\frac{9}{10}$:

$$R(n) = R(\lfloor \tfrac{n-1}{10} \rfloor) + R(\lceil \tfrac{9n-9}{10} \rceil) + \Theta(n)$$

Solução: $R(n)$ é $\Theta(n \lg n)$

Exemplos

Número médio de execuções da linha 4 do PARTICIONE.

Suponha que $A[p \,..\, d]$ é permutação de $1 \,..\, n$.

$A[p \,..\, d]$	execs
1,2	1
2,1	1
média	1

$A[p \,..\, d]$	execs
1,2,3	2+1
2,1,3	2+1
1,3,2	2+0
3,1,2	2+0
2,3,1	2+1
3,2,1	2+1
média	16/6

$A[p\,..\,d]$	execs	$A[p\,..\,d]$	execs
1,2,3,4	3+3	1,3,4,2	3+1
2,1,3,4	3+3	3,1,4,2	3+1
1,3,2,4	3+2	1,4,3,2	3+1
3,1,2,4	3+2	4,1,3,2	3+1
2,3,1,4	3+3	3,4,1,2	3+1
3,2,1,4	3+3	4,3,1,2	3+1
1,2,4,3	3+1	2,3,4,1	3+3
2,1,4,3	3+1	3,2,4,1	3+3
1,4,2,3	3+1	2,4,3,1	3+2
4,1,2,3	3+1	4,2,3,1	3+2
2,4,1,3	3+1	3,4,2,1	3+3
4,2,1,3	3+1	4,3,2,1	3+3
		média	116/24

Quicksort aleatorizado

PARTICIONE-ALEA(A, p, d)
    $i \leftarrow$ RANDOM (p, d)
    $A[i] \leftrightarrow A[d]$
    devolva PARTICIONE (A, p, d)

QUICKSORT-ALE (A, p, d)
    se $p < d$
        então $q \leftarrow$ PARTICIONE-ALEA (A, p, d)
            QUICKSORT-ALE $(A, p, q-1)$
            QUICKSORT-ALE $(A, q+1, d)$

Análise do consumo medio?
Basta contar o número esperado de comparações na linha 4 do PARTICIONE

Consumo de tempo esperado

Suponha $A[p\,..\,r]$ permutação de $1\,..\,n$.

X_{ab} = número de comparações entre a e b na linha 4 de PARTICIONE

Queremos calcular

$$X = \text{total de comparações “}A[j] \le x\text{”} = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} X_{ab}$$

Supondo $a < b$,

$$X_{ab} = \begin{cases} 1 & \text{se primeiro pivô em } \{a, \ldots, b\} \text{ é } a \text{ ou } b \\ 0 & \text{caso contrário} \end{cases}$$

Qual a probabilidade de X_{ab} valer 1?

Consumo de tempo esperado

Supondo $a < b$,

$$X_{ab} = \begin{cases} 1 & \text{se primeiro pivô em } \{a, \ldots, b\} \text{ é } a \text{ ou } b \\ 0 & \text{caso contrário} \end{cases}$$

Qual a probabilidade de X_{ab} valer 1?

$$\Pr\{X_{ab}{=}1\} = \frac{1}{b-a+1} + \frac{1}{b-a+1} = \mathrm{E}[X_{ab}]$$

Consumo de tempo esperado

Supondo $a < b$,

$$X_{ab} = \begin{cases} 1 & \text{se primeiro pivô em } \{a, \ldots, b\} \text{ é } a \text{ ou } b \\ 0 & \text{caso contrário} \end{cases}$$

Qual a probabilidade de X_{ab} valer 1?

$$\Pr\{X_{ab}{=}1\} = \frac{1}{b-a+1} + \frac{1}{b-a+1} = \mathrm{E}[X_{ab}]$$

$$X = \sum_{a=1}^{n-1} \sum_{b=a+1}^{n} X_{ab}$$

$\mathrm{E}[X] =$????

$$
\begin{aligned}
\mathrm{E}[X] &= \sum_{a=1}^{n-1}\sum_{b=a+1}^{n} \mathrm{E}[X_{ab}] \\
&= \sum_{a=1}^{n-1}\sum_{b=a+1}^{n} \Pr\{X_{ab}{=}1\} \\
&= \sum_{a=1}^{n-1}\sum_{b=a+1}^{n} \frac{2}{b-a+1} \\
&= \sum_{a=1}^{n-1}\sum_{k=1}^{n-a} \frac{2}{k+1} \\
&< \sum_{a=1}^{n-1} 2\left(\tfrac{1}{1}+\tfrac{1}{2}+\cdots+\tfrac{1}{n}\right) \\
&< 2n\left(\tfrac{1}{1}+\tfrac{1}{2}+\cdots+\tfrac{1}{n}\right) < 2n\,(1+\ln n)
\end{aligned}
$$

CLRS (A.7), p.1060

Exemplo

1	3	6	2	5	7	4

1	3	2	4	5	7	6

1	2	3	4	5	6	7

	1	2	3	4	5	6	7
1		1	0	1	0	0	0
2	1		1	1	0	0	0
3	0	1		1	0	0	0
4	1	1	1		1	1	1
5	0	0	0	1		1	0
6	0	0	0	1	1		1
7	0	0	0	1	0	1	

Conclusões

O consumo de tempo esperado do algoritmo QUICKSORT-ALE é $\mathrm{O}(n \log n)$.

Do exercício 7.4-4 do CLRS temos que

O consumo de tempo esperado do algoritmo QUICKSORT-ALE é $\Theta(n \log n)$.