QUICKSORT

ANALISI PROBABILISTICA

- COME MERGESORT, ANCHE QUICKSORT E' BASATO SUL PARADIGMA DIVIDE-ET-IMPERA

- PER ORDINARE UN SOTTOARRAY $A[p..r]$, L'ALGORITMO QUICKSORT ESEGUE I TRE PASSAGGI:

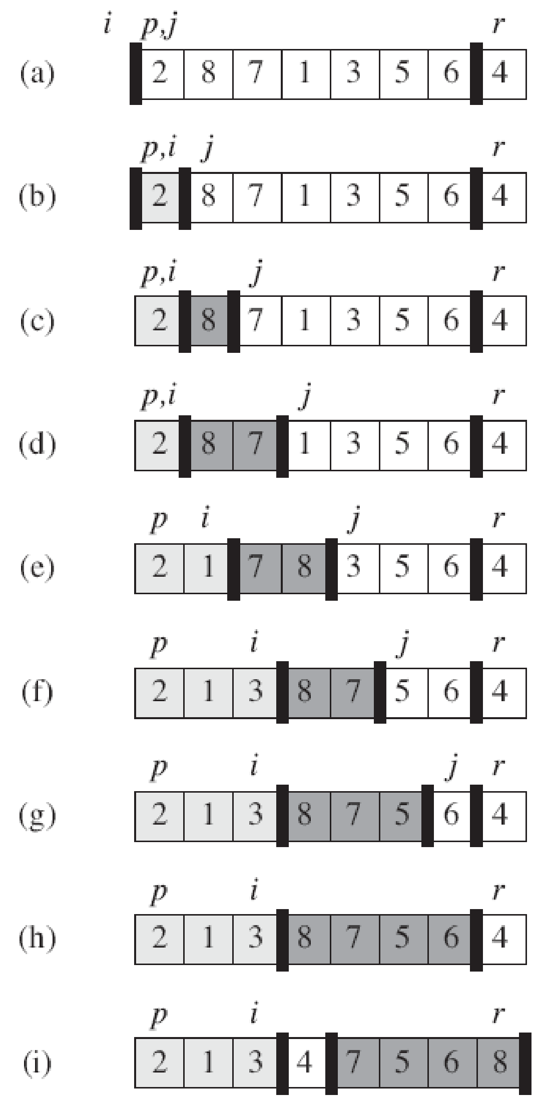
  - DIVIDE: PARTIZIONA $A[p..r]$ IN DUE SOTTOARRAY $A[p..q-1]$ E $A[q+1..r]$ TALI CHE $A[i] \leq A[q] \leq A[j]$, PER OGNI $p \leq i \leq q-1$, $q+1 \leq j \leq r$ CALCOLANDO q (L'ELEMENTO $A[q]$ E' DETTO PIVOT)

  - IMPERA: ORDINA RICORSIVAMENTE $A[p..q-1]$ E $A[q+1..r]$

  - COMBINA: POICHE' QUICKSORT OPERA SUL POSTO, ALLA FINE $A[p..r]$ RISULTA GIA' ORDINATO E NON OCCORRE FARE ALCUN LAVORO

```
QUICKSORT(A, p, r)
if p < r
    q = PARTITION(A, p, r)
    QUICKSORT(A, p, q − 1)
    QUICKSORT(A, q + 1, r)
```

$A[q]$ PIVOT



- PARTIZIONAMENTO SUL POSTO

```
PARTITION(A, p, r)
x = A[r]
i = p − 1
for j = p to r − 1
    if A[j] ≤ x
        i = i + 1
        exchange A[i] with A[j]
exchange A[i + 1] with A[r]
return i + 1
```

COMPLESSITÀ: $\Theta(r-p+1)$

PRESTAZIONI DI QUICKSORT

<u>CASO PEGGIORE</u>: PARTIZIONAMENTO CON SBILANCIAMENTO
(LIMITE INFERIORE) MASSIMO

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

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

SOLUZIONE: $T(n) = \Omega(n^2)$

PRESTAZIONI DI QUICKSORT

CASO PEGGIORE: (LIMITE SUPERIORE)

SI HA: $T(n) = \max_{0 \le q \le n-1} \left(T(q) + T(n-q-1) \right) + \Theta(n)$

DIMOSTRIAMO CHE $T(n) \le c\, n^2$ (CIOE' $T(n) = O(n^2)$)

$$T(n) \le \max_{0 \le q \le n-1} \left(c q^2 + c(n-q-1)^2 \right) + \Theta(n)$$

$$= c \cdot \max_{0 \le q \le n-1} \left(q^2 + (n-q-1)^2 \right) + \Theta(n)$$

$$\le c \cdot n^2 + \Theta(n) = O(n^2)$$

- POICHE' LA FUNZIONE $q^2 + (n-q-1)^2$ E' CONVESSA, ESSA RAGGIUNGE IL $\max$ IN UN ESTREMO, CIOE' PER $q = 0$ OPPURE $q = n-1$.

QUINDI: $q^2 + (n-q-1)^2 \leq (n-1)^2$

- PERTANTO:

$$
\begin{aligned}
T(n) &\leq c \cdot \max_{0 \leq q \leq n-1} \left(q^2 + (n-q-1)^2 \right) + \Theta(n) \\
&\leq c \cdot (n-1)^2 + \Theta(n) \\
&= c\, n^2 - c(2n-1) + \Theta(n) \\
&\leq c\, n^2 \qquad \text{(PER } c \text{ SUFFICIENTEMENTE GRANDE)}
\end{aligned}
$$

- DUNQUE, PER INDUZIONE, $T(n) = O(n^2)$.

PRESTAZIONI DI QUICKSORT

CASO MIGLIORE: PARTIZIONAMENTO CON BILANCIAMENTO MASSIMO

$$T(n) = 2T\left(\frac{n}{2}\right) + \Theta(n)$$

SOLUZIONE: $T(n) = \Theta(n \lg n)$

UNA VERSIONE RANDOMIZZATA DI QUICKSORT

```
RANDOMIZED-PARTITION(A, p, r)
    i = RANDOM(p, r)
    exchange A[r] with A[i]
    return PARTITION(A, p, r)
```

A[i] PIVOT

```
RANDOMIZED-QUICKSORT(A, p, r)
    if p < r
        q = RANDOMIZED-PARTITION(A, p, r)
        RANDOMIZED-QUICKSORT(A, p, q - 1)
        RANDOMIZED-QUICKSORT(A, q + 1, r)
```

- LE ANALISI NEI CASI MIGLIORE E PEGGIORE DI RANDOMIZED-QUICKSORT SONO IDENTICHE A QUELLE VISTE PRIMA PER QUICKSORT

ANALISI NEL CASO MEDIO

- SIA X IL NUMERO COMPLESSIVO DI CONFRONTI TRA ELEMENTI DI UN ARRAY DI DIMENSIONE n I CUI ELEMENTI SONO A DUE A DUE DISTINTI
- ALLORA IL TEMPO DI ESECUZIONE DI RANDOMIZED-QUICKSORT È $O(n+X)$
- OCCORRE VALUTARE $E[X]$
- SIANO $z_1 < z_2 < \dots < z_n$ GLI ELEMENTI DELL'ARRAY A
- PONIAMO $Z_{ij} = \{z_i, z_{i+1}, \dots, z_j\}$, PER $1 \le i \le j \le n$
- OGNI COPPIA DI ELEMENTI VIENE CONFRONTATA AL MASSIMO UNA VOLTA

- PONIAMO

$$X_{ij} = I\{z_i \text{ È CONFRONTATO CON } z_j\}$$

QUINDI $$X = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}$$

- $$E[X] = E\left[\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} X_{ij}\right]$$

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} E[X_{ij}]$$

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} Pr\{z_i \text{ È CONFRONTATO CON } z_j\}$$

- <u>OSSERVAZIONE</u>: z_i E' CONFRONTATO CON z_j SE E SOLO SE IL PRIMO ELEMENTO DI Z_{ij} CHE VIENE SCELTO COME PIVOT E' z_i O z_j

$Pr\{z_i$ E' CONFRONTATO CON $z_j\} =$

$= Pr\{z_i$ O z_j E' IL PRIMO PIVOT SCELTO DA $Z_{ij}\}$

$= Pr\{z_i$ E' IL PRIMO PIVOT SCELTO DA $Z_{ij}\}$

$+ Pr\{z_j$ E' IL PRIMO PIVOT SCELTO DA $Z_{ij}\}$

$$= \frac{1}{j-i+1} + \frac{1}{j-i+1} = \frac{2}{j-i+1}$$

- DUNQUE

$$E[X] = \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \Pr\{z_i \text{ È CONFRONTATO CON } z_j\}$$

$$= \sum_{i=1}^{n-1} \sum_{j=i+1}^{n} \frac{2}{j-i+1}$$

$$= \sum_{i=1}^{n-1} \sum_{k=1}^{n-i} \frac{2}{k+1}$$

$$< \sum_{i=1}^{n-1} \sum_{k=1}^{n-i} \frac{2}{k}$$

$$= \sum_{i=1}^{n-1} O(\lg n)$$

$$= O(n \lg n) \qquad \blacksquare$$

ESERCIZIO

QUAL E' IL TEMPO DI ESECUZIONE DI QUICKSORT QUANDO

- TUTTI GLI ELEMENTI HANNO LO STESSO VALORE?
- L'ARRAY A CONTIENE ELEMENTI A DUE A DUE DISTINTI ED E' GIA' ORDINATO?