

Alcuni esercizi sulla tecnica del Divide et Impera.

1. a) Descrivere gli aspetti essenziali della tecnica Divide et Impera, utilizzando lo spazio designato.
b) Descrivere ed analizzare un algoritmo **basato sulla tecnica Divide et Impera** che dato un array $A[1, \dots, n]$ di interi ne restituisca il massimo.
2. Sia $V[1..n]$ un vettore ordinato di 0 e 1.
Descrivere ed analizzare un algoritmo per determinare il numero di 0 presenti in $V[1..n]$ in tempo $O(\log n)$.
3. Sia $A[1, \dots, n]$ un vettore ordinato che e' stato shiftato k posizioni a destra. Ad esempio, il vettore $[35, 42, 1, 7, 15, 18, 28, 30]$ e' un vettore ordinato che e' stato shiftato $k = 2$ posizioni a destra, mentre il vettore $[15, 18, 28, 30, 35, 42, 1, 7]$ e' un vettore ordinato che e' stato shiftato $k = 6$ posizioni a destra.
(a) Supponendo di avere A e k in input, descrivere un algoritmo che determini il massimo in A in tempo $O(1)$.
(b) Supponendo di avere solo il vettore A in input, descrivere un algoritmo che determini il massimo in A in tempo $O(\log n)$.
4. a) Si descrivano (soltanto) l'input e l'output della procedura PARTITION studiata.
b) Descrivere un algoritmo che, dati un array $A[1..n]$ di interi positivi distinti e un intero k con $1 \leq k \leq n$, calcoli il k -esimo minimo di A . Tale algoritmo non dovra' fare alcun ricorso ad algoritmi di ordinamento, ma potra' utilizzare la procedura PARTITION.
E' necessario spiegare verbalmente il funzionamento dell'algoritmo proposto, indicare la tecnica di progettazione usata e giustificare la correttezza dell'algoritmo.
Esempio: se $A[1..7] = [6, 9, 3, 2, 5, 7, 8]$ e $k = 3$ l'algoritmo dovra' restituire 5.
c) Analizzare il caso peggiore dell'algoritmo proposto al punto b).
5. a) Fornire lo pseudocodice di un algoritmo ricorsivo che ordini un array $A[1..n]$ nel seguente modo: prima ordina ricorsivamente $A[1..n-1]$ e poi inserisce $A[n]$ nell'array ordinato $A[1..n-1]$.
b) Analizzare la complessita' di tempo dell'algoritmo proposto al punto b).
6. Sia dato un vettore binario ordinato $A[1..n]$.
(a) Progettare un algoritmo di complessita' $\Theta(n)$ nel caso peggiore, che conti il numero di occorrenze di 1 nel vettore A .
(b) Progettare un algoritmo di complessita' $O(\log n)$, che conti il numero di occorrenze di 1 nel vettore A .
7. Sia dato un vettore ordinato $A[1..n]$ di interi distinti. Progettare un algoritmo che determini, in tempo $O(\log n)$, se esiste o meno un intero i tale che $A[i] = i$.
8. Descrivere ed analizzare un algoritmo basato sul paradigma *divide et impera* che dato un vettore *ordinato* $A[1..n]$ di interi strettamente positivi (cioe' per ogni $1 \leq i \leq n$, $A[i] \geq 1$), restituisca il numero di occorrenze di 1 nel vettore A . L'algoritmo deve avere complessita' di tempo $O(\log n)$.
9. Descrivere ed analizzare un algoritmo che presi in input un vettore $A[1..n]$ di interi distinti e un intero i , $2 \leq i \leq n$, tale che $A[1..i-1]$ e' ordinato in ordine crescente, inserisca $A[i]$ nell'ordine che gli compete fra i suoi predecessori, ovvero modifichi A in maniera che dopo l'esecuzione $A[1..i]$ contenga gli stessi elementi di prima, ma in ordine crescente, e i restanti elementi di A siano immutati. L'algoritmo deve avere una complessita' di tempo $O(\log n)$.

Per esempio: se $A[1..6] = [2, 4, 7, 3, 8, 1]$ ed $i = 4$, al termine dell'esecuzione il vettore risultante sarà $A[1..6] = [2, 3, 4, 7, 8, 1]$.

10. Progettare ed analizzare un algoritmo basato sulla tecnica Divide et Impera che, dato un vettore $A[1..n]$, con $A[i] \in \{0, 1\}$, per ogni $i = 1, \dots, n$, restituisca VERO se esiste $i \in \{1, \dots, n-1\}$ tale che $A[i] = 1$ e $A[i+1] = 0$, FALSO altrimenti.
11. Si consideri il problema di determinare simultaneamente il minimo ed il massimo in un vettore di interi. Si descriva un algoritmo basato sulla tecnica Divide et Impera che, dato un vettore $A[1..n]$ di interi, restituisca una coppia (min, max) , dove min è il valore minimo in A e max il suo valore massimo. Analizzare il tempo di esecuzione.
12. Sia dato un vettore di n elementi che possono assumere solo i tre valori VERDE, BIANCO e ROSSO. Descrivere ed analizzare un algoritmo basato sulla tecnica Divide-et Impera che decida se nel vettore vi sono 3 occorrenze consecutive di ROSSO.
13. Progettare ed analizzare un algoritmo basato sulla tecnica Divide et Impera che, dato un vettore *ordinato* $A[1..n]$, ed un intero k , restituisca VERO se esiste $i \in \{1, \dots, n\}$ tale che $A[i] > k$, FALSO altrimenti, in tempo $O(\log n)$.