

Esercizi Divide-et-impera

- ➊ **Punto fisso:** Scrivere un algoritmo che prenda in input un vettore ordinato A contenente n interi distinti e restituisca **true** se e solo se esiste un indice i tale che $A[i] = i$, in tempo $O(\log n)$.
- ➋ **Chi manca?:** Scrivere un algoritmo che prenda in input un vettore ordinato $A[1 \dots n]$ contenente n elementi interi distinti appartenenti all'intervallo $1 \dots n + 1$ e restituisca in tempo $O(\log n)$ l'unico intero dell'intervallo $1 \dots n + 1$ che non compare in A .
- ➌ **Vettori uni-modulari:** Un vettore di interi distinti A è detto **unimodulare** se esiste un indice h tale che $A[1] > A[2] > \dots > A[h-1] > A[h]$ e $A[h] < A[h+1] < A[h+2] < \dots < A[n]$, dove n è la dimensione del vettore. Scrivere un algoritmo che prenda in input un vettore unimodulare e restituisca il valore minimo del vettore in tempo $O(\log n)$.

Discutere correttezza e complessità degli algoritmi proposti.

Samarcanda

Nel gioco di Samarcanda, ogni giocatore è figlio di una nobile famiglia della Serenissima, il cui compito è di partire da Venezia con una certa dotazione di denari, arrivare nelle ricche città orientali, acquistare le merci preziose al prezzo più conveniente e tornare alla propria città per rivenderle.

Scrivere un algoritmo che prenda in input un vettore P contenente n interi in cui $P[i]$ è il prezzo di una certa merce al giorno i e restituisca il guadagno massimo $P[y] - P[x]$ che si può ottenere comprando la merce nel giorno x e rivendendola il giorno y , con $x < y$.

Discutere correttezza e complessità dell'algoritmo proposto.

Per fare un albero (binario di ricerca) ci vuole...

Dato un vettore V contenente n interi ordinati e distinti, scrivere un algoritmo che restituisca un albero binario di ricerca di altezza minima.

Discutere correttezza e complessità dell'algoritmo proposto.

Maggioranza

Scrivere una funzione

```
boolean hasMajority(int[] A, int n)
```

che prenda in input un vettore ordinato *A* contenente *n* interi e restituisca **true** se *A* contiene un valore di maggioranza, ovvero un valore che compare più di $n/2$ volte; restituisca **false** altrimenti. Soluzioni di costo computazionale lineare non verranno prese in considerazione.

Discutere correttezza e complessità dell'algoritmo proposto.