

PRIMA PARTE

1. **[2 punti]** Sia $\mathcal{A}$ un algoritmo per un problema computazionale Π . Si supponga che per ogni n esista un'istanza di taglia n che richiede $\geq n^2$ operazioni. Cos'altro si deve provare per dimostrare che la complessità al caso pessimo di $\mathcal{A}$ è $\Theta(n^2)$?
2. **[4 punti]** Dare la definizione di *albero binario completo* e dimostrare che un albero binario completo con n nodi ha altezza $h = \lfloor \log_2 n \rfloor$.
3. **[3 punti]** Nell'inserimento di una nuova entry in un (2,4)-Tree un nodo v può andare in *overflow*
 - (a) Cosa si intende per overflow di v ?
 - (b) Descrivere brevemente la procedura $\text{Split}(v)$ usata per sanare la condizione di overflow analizzandone la complessità.
4. **[3 punti]** Spiegare cosa significa che Randomized QuickSort ha complessità $O(n \log n)$ con alta probabilità.
5. **[4 punti]** Sapendo che un free tree con n vertici ha $m = n - 1$ archi, dimostrare che un grafo connesso con n vertici ha $m \geq n - 1$ archi.

SECONDA PARTE

1. [5 punti] Sia T un testo di lunghezza n e P un pattern di lunghezza $m < n$. Il seguente algoritmo è una variante di `BruteForceMatch( $T, P$ )`, che usa un unico ciclo `while` e scansiona T e P da destra verso sinistra invece che viceversa:

```

i ← n-m; j ← m-1;
while ((i ≥ 0) AND (j ≥ 0)) do {
    if (T[i+j] = P[j]) then {j ← j-1}
    else {i ← i-1; j ← m-1}
}
if (j=-1) then return i else return -1

```

- (a) Trovare un opportuno invariante per il ciclo **while**
- (b) Identificare un'istanza (T, P) per la quale l'algoritmo esegue $\Theta((n-m)m)$ operazioni
2. [6 punti] Sia T un (2,4)-Tree dove ogni nodo $v \in T$ memorizza in una variabile $v.size$ il numero di entry in T_v (incluse quelle in v). Progettare un algoritmo non ricorsivo che conti quante entry in T hanno chiave $\leq k$, e analizzarne la complessità.
3. [5 punti] Sia $G = (V, E)$ un grafo non diretto e connesso in cui ciascun vertice ha grado esattamente c , con $c > 2$ costante intera. Si consideri l'esecuzione di $BFS(G, s)$ a partire da un arbitrario vertice $s \in V$. Dimostrare per induzione su i che il livello L_i generato da $BFS(G, s)$ contiene $\leq c \cdot (c-1)^{i-1}$ vertici, per ogni $i \geq 0$.

TEMPO COMPLESSIVO A DISPOSIZIONE: 2.5 ore