

PRIMA PARTE

1. **[3 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]** Dimostrare che un albero binario completo con n nodi ha altezza $h = \lfloor \log_2 n \rfloor$.
3. **[5 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 un nodo?
 - (b) Descrivere brevemente la procedura Split(v) usata per sanare la condizione di overflow analizzandone la complessità.
4. **[4 punti]** Si consideri un grafo $G = (V, E)$.
 - (a) Dare la definizione di *spanning forest* di G .
 - (b) Si supponga che G abbia 3 componenti connesse, G_1, G_2 e G_3 . Detti n_i il numero di vertici e m_i il numero di archi di G_i , siano: $n_1 = 5, m_1 = 10$; $n_2 = 6, m_2 = 15$; $n_3 = 8, m_3 = 8$. Qual è il numero massimo di archi in una spanning forest di G ? Motivare la risposta.

SECONDA PARTE

1. [5 punti] Sia A una matrice $n \times m$ con valori interi, con la proprietà che i valori in ogni colonna sono ordinati in senso decrescente dalla riga 0 alla riga $n - 1$. Il seguente algoritmo determina il massimo numero di valori > 0 in una colonna.

```
i ← 0; j ← 0;
while (i < n) AND (j < m) do{
    if (A[i,j] ≤ 0) then j ← j + 1;
    else i ← i + 1;
}
return i;
```

Trovare un opportuno invariante per il ciclo, che serva per provare la correttezza dell'algoritmo (la prova di correttezza non è richiesta).

2. [6 punti] Sia T un albero binario di ricerca le cui entry rappresentano studenti di un'università. Ogni studente è associato a una entry (k, x) , dove k è la matricola, e x indica se lo studente è straniero ($x = 1$) o italiano ($x = 0$). Per ogni nodo $v \in T$ esiste un intero $v.\text{numStr}$ che riporta il numero di studenti stranieri in T_v (sottoalbero con radice v). Si vuole progettare un algoritmo ricorsivo **MinMatStraniero** per determinare *la più piccola matricola di uno studente straniero in T* . Se non ci sono studenti stranieri in T_v l'algoritmo restituisce **null**.
- (a) Descrivere tramite pseudocodice la generica invocazione di **MinMatStraniero**, specificandone con attenzione l'input e l'output.
 - (b) Analizzare la complessità di **MinMatStraniero**
3. [5 punti] Sia S una sequenza di n chiavi distinte $\in [0, 10n]$. Progettare un algoritmo di complessità lineare che restituisca l'indice i della *mediana*, cioè tale che esistono esattamente $\lfloor n/2 \rfloor$ chiavi più piccole di $S[i]$ in S .