

Tema d'esame di Algoritmi e Strutture Dati

Modulo B

18/06/2001

Tempo a disposizione: 3 ore.

Considerate il problema di memorizzare una collezione di elementi tramite un albero di ricerca binario in modo da minimizzare il numero di confronti necessari per trovare un dato elemento. Se tutti gli elementi avessero uguale probabilità di essere ricercati, allora la soluzione ottima per memorizzare gli elementi sarebbe tramite un albero perfettamente bilanciato. Se però alcuni elementi vengono ricercati con maggiore frequenza di altri, un albero non bilanciato può essere preferibile.

Sopponete di avere N elementi distinti con chiavi $k_1 < k_2 < \dots < k_N$, e che l'elemento i -esimo abbia probabilità $p_i \geq 0$ di essere ricercato. Supponete, inoltre, che ogni ricerca acceda effettivamente ad un elemento nell'albero, in altre parole che valga $\sum_{i=1}^N p_i = 1$. Se l'elemento i -esimo è memorizzato a livello l_i dell'albero (la radice è a livello 1, i suoi figli a livello 2, ecc.), allora per trovare l'elemento i saranno necessari l_i confronti.

Quindi, per un dato albero, il numero medio di confronti necessari per effettuare una ricerca è $\sum_{i=1}^N p_i l_i$. L'obiettivo è determinare l'albero binario che minimizza tale quantità.

1. Considerate l'algoritmo "greedy" che inserisce le chiavi in ordine decrescente di frequenza. Fornire un esempio minimo (con il più piccolo numero di elementi) che mostri che questa strategia greedy non sempre fornisce l'albero ottimo di ricerca.
2. Fornire un algoritmo che risolve correttamente il problema, e se ne analizzi la complessità.