

008AA – ALGORITMICA E LABORATORIO

Verifica del 21 giugno 2011

Cognome Nome:

N. Matricola:

Corso: A B

Esercizio 1. (*2+4 punti*) Sia data una tabella hash di dimensione $m = 13$ in cui le collisioni sono gestite con indirizzamento aperto e hash doppio. Progettare la funzione che restituisce la sequenza di probing, e la si usi per inserire nella tabella hash le seguenti chiavi $S = \{3, 13, 20, 15, 7, 1, 6, 18, 4\}$.

Esercizio 2. (*6+2 punti*) Sia dato un albero binario T di radice r , in cui ciascun nodo u memorizza un numero intero nel campo **chiave**. Si progetti un algoritmo ricorsivo che stampa le chiavi contenute nei nodi u che soddisfano la condizione: la somma delle chiavi contenute negli antenati di u (incluso) è uguale alla somma delle chiavi contenute nei discendenti di u (incluso). Si determini inoltre la complessità in tempo al caso pessimo dell'algoritmo proposto.

Esercizio 3. (*6 punti*) Sia data una matrice M quadrata $n \times n$ di interi positivi, le cui righe e colonne sono numerate da 1 a n . Un cammino su tale matrice è una sequenza di n celle che parte da una cella della colonna 1 e finisce su una cella della colonna n . Un cammino può attraversare solo celle "adiacenti" che sono definite come segue: $M[i, j]$ è adiacente a $M[i, j + 1]$, $M[i - 1, j + 1]$ e $M[i + 1, j + 1]$, quindi alle celle nella colonna successiva che condividono il lato destro o uno spigolo con $M[i, j]$. Il peso di un cammino è la somma degli interi contenuti nelle celle che lo costituiscono.

Si progetti un algoritmo basato sulla tecnica della Programmazione Dinamica che calcola, per ogni cella dell'ultima colonna $M[i, n]$, il peso minimo di un cammino che arriva in quella cella.

Esercizio 4. (*5+5 punti*) Sia dato il grafo pesato e non orientato $G = (V, E)$ avente nodi etichettati con lettere e archi $E = \{(A, B; 4), (B, C; 21), (A, C; 14), (A, D; 30), (C, D; 10), (C, E; 1), (E, F; 6), (F, G; 4), (E, G; 9)\}$ (il numero dopo il punto-e-virgola indica il peso dell'arco). Si illustri il funzionamento degli algoritmi di Prim e Kruskal su G dettagliando la coda di priorità nell'uno e le fusioni nell'altro.