
Algoritmi e Strutture di Dati – A.A. 2010-2011
Seconda Prova dell'appello dell'14-15 giugno 2011
Libri e appunti chiusi

Studente 270/04 (tempo prova = 2:15 h) Studente 509/99 (tempo prova = 2:00 h)

Ho bisogno di una correzione veloce in quanto _____

Cognome: _____ **Nome:** _____ **Matricola:** _____

PSEUDOCODIFICA

Negli esercizi seguenti supponi che i grafi siano diretti, rappresentati con liste di adiacenza (ogni grafo è un array di liste doppiamente concatenate).

Esercizio 1

Scrivi lo pseudocodice della procedura NO-ARCHI-DA- $A(u,v)$ che prende in input il grafo A e l'indice di due nodi u e v e restituisce **true** se non c'è l'arco da u a v , **false** altrimenti.

Esercizio 2

Scrivi lo pseudocodice della procedura AGGIUNGI-ARCHI-OPPOSTI(A,u) che prende in input il grafo A e l'indice di un nodo u e che, per ogni arco uscente da u ed entrante in un nodo v , aggiunge anche l'arco opposto da v ad u qualora sia assente (scrivi anche la procedura di aggiunta di un elemento in una lista se necessaria).

Esercizio 3

Scrivi lo pseudocodice della procedura GRAFO-INDIRETTO(A) che prende in input il grafo A , che in generale è un grafo diretto, e lo trasforma in un grafo indiretto, introducendo per ogni arco (u,v) l'arco opposto (v,u) se assente.

Esercizio 4 (solo studenti D.M. 270/04)

Scrivi lo pseudocodice della procedura CONTA-RAGGIUNTI(A,u) che prende in input il grafo A e l'indice di un nodo u e produce in output il numero dei nodi raggiungibili da u (u compreso). Un nodo v è raggiungibile da u se esiste un cammino diretto da u a v .

Esercizio 5

Discuti la complessità computazionale (nel solo caso peggiore) delle procedure che hai proposto per gli esercizi precedenti, utilizzando n per denotare il numero totale dei nodi ed m per denotare il numero totale degli archi del grafo. Ipotizza prima che il grafo sia sparso, cioè $m \in O(n)$, e poi che il grafo sia denso, cioè $m \in O(n^2)$.

CODIFICA C

Si consideri una sequenza di 6 numeri da 1 a 90, ordinati dal più piccolo al più grande, che rappresentano una giocata (o un'estrazione) al superenalotto. Definire le strutture dati più adeguate e implementare i seguenti metodi:

ESERCIZIO 1) Data una sequenza estratta E e una lista di sequenze giocate S, contare quante di queste sono vincenti. Una sequenza è vincente se indovina almeno tre numeri della sequenza estratta.

Esempio:

Dati E: **[1,4,16,18,32,80]** ed S: [40,50,60,70,**80**,90]->**[2,4,6,8,10,12]**->**[1,2,4,8,16,32]**

La lista S contiene una sola sequenza vincente (l'ultima, in quanto indovina i numeri **1,4,16,32**)

ESERCIZIO 2) Data una sequenza estratta E e una lista di sequenze giocate S, rimuovere tutte le sequenze non vincenti (che non indovinano almeno tre numeri).

Esempio:

Dati E: **[1,4,16,18,32,80]** ed S: [40,50,60,70,**80**,90]->**[2,4,6,8,10,12]**->**[1,2,4,8,16,32]**

La funzione deve modificare S come segue:

S: **[1,2,4,8,16,32]**