
Algoritmi e Strutture di Dati – A.A. 2009-2010
Seconda Prova appello del 15 giugno 2010
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 si fa riferimento a grafi diretti rappresentati con liste di adiacenza (ogni grafo è un array di liste doppiamente concatenate).

Esercizio 1

Scrivi lo pseudocodice della procedura VERIFICA-NO-ARCHI(A) che prende in input il grafo A e restituisce **true** se il grafo non ha nemmeno un arco, **false** altrimenti.

Esercizio 2

Scrivi lo pseudocodice della procedura VERIFICA-NODO-ISOLATO(A,u) che prende in input il grafo A e l'indice u di un nodo e restituisce **true** se il nodo è isolato dal resto del grafo (cioè non ha archi uscenti né entranti), **false** altrimenti.

Esercizio 3

Scrivi lo pseudocodice della procedura INVERTI-ARCHI(A) che prende in input il grafo A e che, senza modificare A, restituisce in output un nuovo grafo B che ha tutti gli archi diretti in modo inverso (cioè, il grafo B ha un arco da un nodo con indice u ad un nodo con indice v se e solo se il grafo A ha un arco dal nodo v al nodo u).

Esercizio 4 (solo studenti 270/04)

Scrivi lo pseudocodice della procedura ESISTE-NODO-ISOLATO(A) che prende in input il grafo A e restituisce **true** se il grafo ha almeno un nodo isolato, **false** altrimenti. N.B.: è possibile implementare questa procedura in un tempo strettamente minore di $O(n^3)$, se lo sai fare, fallo.

Esercizio 5

Discuti la complessità computazionale (nel solo caso peggiore) delle procedure che hai proposto relativamente agli esercizi precedenti.

CODIFICA C

ESERCIZIO 1) Si consideri uno stradario costituito da un elenco di "vie". Ogni via è contraddistinta da un codice univoco (intero) e un nome (stringa di massimo 20 caratteri).

Per esempio: {**10**, "Vasca Navale"}, {**22**, "Corrado Segre"}, ...

Vogliamo gestire un elenco di persone: ogni persona ha un codice univoco (intero), una via (intero, corrispondente ad un codice dello stradario) e il numero della via (un intero).

Per esempio: {50, **10**, 16}, {63, **10**, 18}, {13, **22**, 4}, ...

Definire:

- Strutture dati adeguate per gestire in modo dinamico sia l'elenco di vie che l'elenco di persone.
- Un metodo che dato lo stradario, l'elenco di persone, e un nome di una via (stringa di massimo 20 caratteri), stampi su video i codici di tutte le persone associate alla stessa via (codice corrispondente all'indirizzo di input).

Per esempio: dati gli elenchi dei due esempi precedenti e come nome "Vasca Navale" il programma stampa su video i codici 50 e 63.

- Un metodo che dato un codice di una via, elimini dallo stradario la via in questione e dall'elenco delle persone tutti coloro che sono associati (tramite il codice) a quella via.

Per esempio: dati gli elenchi dei due esempi iniziali e come codice di via **10**, il metodo elimina {**10**, "Vasca Navale"} dallo stradario e {50, **10**, 16}, {63, **10**, 18} dall'elenco delle persone.

ESERCIZIO 2) Dato il seguente codice

```
#include <stdio.h>

int f(int a[], int n) {
    int i = 0;
    int cont = 0;
    while(i < n) {
        if (!(i % 2)) {
            cont += a[i];
        }
        i++;
    }
    return cont;
}
```

Descrivere a parole la funzionalità realizzata dalla funzione f .

Fornire un esempio di output su un input a piacimento.

Codificare una versione ricorsiva di f.