



Esercitazione di laboratorio n. 10

Laboratorio **NON** valutato

Caricamento nella Sezione Elaborati del Portale entro e non oltre le 23:59 del 15/12/2015

Esercizio n. 1: Nonogram

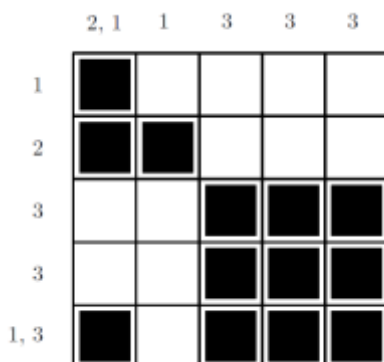
Tema d'esame del 02/09/2015

I "Nonogram" oppure "Paint by Numbers" (dipingere con i numeri) o "Griddlers", sono dei rompicapo logico-grafici in cui le celle di una griglia $N \times M$ devono essere colorate o lasciate in bianco in base a dei numeri a lato della griglia.

In questo tipo di rompicapo, ogni numero indica quante celle consecutive devono essere riempite in una riga o in una colonna. Per esempio, un'etichetta di riga/colonna del tipo "4 8 3" significa che su quella riga/colonna ci sono 3 insiemi di 4, 8 e 3 celle nere rispettivamente da riempire in questo ordine, con almeno una cella bianca tra gruppi successivi di celle nere.

Il gioco consiste nell'individuare una configurazione di celle nere congruente con l'etichettatura delle righe e delle colonne.

La figura a seguire mostra una griglia per nonogram con una soluzione valida per l'etichettatura di righe e colonne proposta.



Un primo file di testo il cui nome è passato sulla riga di comando contiene:

- nella prima riga del file il numero di righe N della griglia
- per ogni riga della griglia un gruppo di informazioni (un gruppo per riga del file):
 - numero di insiemi n_i di celle nere contigue per quella riga della griglia
 - n_i interi che rappresentano la lunghezza di ciascuno di questi insiemi
- segue su una riga del file il numero di colonne M della griglia
- per ogni colonna della griglia un gruppo di informazioni (un gruppo per riga del file):
 - numero di insiemi n_i di celle nere contigue per quella colonna della griglia
 - n_i interi che rappresentano la lunghezza di ciascuno di questi insiemi.

Si riporta di seguito il file relativo all'esempio di cui sopra:

```
5
1 1
1 2
1 3
1 3
```



2 1 3
5
2 2 1
1 1
1 3
1 3
1 3

Si scriva un programma C che:

1. legga i vincoli dal primo file di testo e li memorizzi in un'opportuna struttura dati in memoria
2. calcoli in modo ricorsivo una soluzione del gioco e la scriva su un secondo file di testo il cui nome sia passato sulla riga di comandi. Il formato di questo file sia a discrezione del programmatore.

Esercizio n. 2: Gruppo di acquisto, con ADT

Si progetti una struttura dati adatta a memorizzare le informazioni relative ad una sequenza di liste della spesa associate ad un gruppo di acquisto.

Le informazioni relative ai prodotti acquistabili sono riportate in un file testuale (`prodotti.txt`), il quale deve essere a sua volta caricato in memoria. Il file riporta l'insieme di diversi prodotti disponibili, indicando per ogni prodotto il nome (stringa di massimo 30 caratteri senza spazi) e il prezzo per unità di prodotto (float).

Un secondo file testuale (`liste.txt`) contiene la descrizione di un insieme di liste della spesa, che fanno riferimento ai prodotti di cui sopra. La prima riga del file riporta il numero totale di liste contenute. Nelle righe successive, per ogni lista della spesa è poi specificato il numero P di prodotti che la compongono e **il nome della lista** (*si noti che questa è un'aggiunta rispetto alle versioni precedenti*), seguito da P righe riportanti il nome e la quantità di ogni prodotto incluso.

Si scriva un programma in linguaggio C che permetta di portare in memoria tutte le informazioni di cui sopra, e renda disponibili le seguenti operazioni:

- stampa a video dei dettagli di tutte le liste
- calcolo del costo per ogni lista della spesa
- calcolo del costo totale per tutti gli acquisti del gruppo
- aggiunta di una nuova lista della spesa
- aggiunta di un prodotto a una certa lista della spesa (*identificata mediante il nome*)

Le strutture dati siano implementate come ADT di prima categoria: uno per la tabella dei prodotti e uno per l'elenco delle liste della spesa.

Per quanto riguarda l'elenco dei prodotti, che sono in numero immutabile, si realizzi un ADT tabella di simboli implementata utilizzando un vettore (di prodotti).

L'elenco di liste della spesa sia invece realizzato mediante un ADT coda (lista concatenata). Tale modulo si faccia carico anche della gestione dei contenuti delle singole liste della spesa, ossia non è richiesto un terzo ADT che gestisca l'elenco di prodotti in una data lista.

A differenza di quanto richiesto nei Lab06 e Lab07, anziché utilizzare riferimenti puntati a un prodotto, si memorizzi l'indice ad esso associato: ogni prodotto viene quindi identificato univocamente da un indice intero nella tabella dei prodotti.

La figura sotto riportata rappresenta la struttura richiesta.

