

Programmazione ad Oggetti - parte A: Esercitazione di laboratorio 4

Esercizio 1:

Scrivere un programma in grado di fondere due liste singolarmente concatenate di interi L1, L2 (non ordinate) in una singola lista L3 ordinata in modo decrescente.

Suggerimento: scrivere una funzione “fondiliste” che utilizzi al suo interno la funzione `insert_ordered` vista a lezione e che accetti come parametri in ingresso i riferimenti alle liste L1, L2, L3.

Esempio: L1= {5, 1, 3}, L2= {10, 12, 7, 2}, L3= {12, 10, 7, 5, 3, 2, 1}

Esercizio 2:

Definire una lista concatenata composta da oggetti Item che contengono ciascuno due stringhe: una stringa abbreviata di due lettere (ad esempio TO, MI, PR) e un stringa che contiene la parola estesa (ad esempio “Torino”, “Milano”) corrispondente.

Scrivere una funzione che ricevendo come parametri un riferimento alla lista, una abbreviazione e la corrispondente parola estesa cerchi l’abbreviazione nella lista e

- a) restituisca 0 se esiste un elemento che possiede la parola estesa e l’abbreviazione
- b) restituisca 1 se non esiste un elemento che possiede la parola estesa e l’abbreviazione e inserisca un nuovo Item in coda alla lista che contenga la parola estesa e l’abbreviazione
- c) restituisca 2 se l’abbreviazione è presente ma ad essa corrisponde una parola differente. La parola estesa dell’elemento della lista deve essere sostituita da quella passata come parametro.

Esercizio 3:

Scrivere un programma che crei un array di liste di dimensione N. Le liste contengono Item con valori interi positivi. Il programma deve calcolare l’indice dell’array contenente la lista in cui compare l’elemento di valore massimo tra gli elementi di tutte le liste.

Esercizio 4:

Scrivere un programma che crei una lista contenente oggetti Item con valori interi.

Si scriva una funzione che ricevendo in ingresso un riferimento alla lista la modifichi, memorizzando nell’ultimo nodo il prodotto del penultimo ed ultimo nodo, nel penultimo il prodotto del terzultimo e del penultimo e così via. Il primo elemento della lista non deve essere modificato.

Ad esempio, una lista contenente la sequenza di interi 4 6 2 3 9 verrà modificata dalla funzione nella lista 4 24 12 6 27.