





Facoltà di Scienze  
Matematiche  
Fisiche Naturali

## Laboratorio di Algoritmi e Strutture Dati

Prof. Aniello Murano

Esercitazione di laboratorio su  
Liste Puntate Semplici

Corso di Laurea  
Codice insegnamento  
Email docente  
Anno accademico

Informatica  
13917  
murano@na.infn.it  
2007/2008

Lezione numero: 7  
Parole chiave: **Liste dinamiche**

next





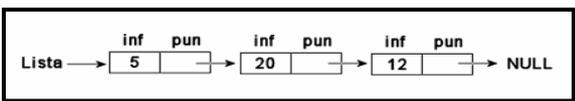

27/09/2007



Facoltà di Scienze  
Matematiche  
Fisiche Naturali

## Liste puntate (remind)

Una lista puntata (semplice) ha una gestione sequenziale: gli elementi si possono scorrere in un unico verso. Inoltre, nella lista è sempre possibile individuare un primo elemento (testa) e un ultimo elemento (coda)



```

Lista → [ inf | pun ] → [ inf | pun ] → [ inf | pun ] → NULL
         |   |         |   |         |   |
         5   →         20  →         12  →
    
```

back

✖

next

Federica 27/09/2007 3 Facoltà di Scienze Matematiche Fisiche Naturali

## Operazioni sulle liste

**Nella precedente lezione abbiamo visto l'implementazione delle seguenti operazioni su liste**

- Inizializzazione
- Inserimento in coda
- Verifica lista vuota
- Stampa lista

**Oggi implementeremo inoltre le seguenti operazioni**

- Ricerca elemento
- Inserimento elemento
- Cancellazione di un elemento

back ✖ next

Federica 27/09/2007 4 Facoltà di Scienze Matematiche Fisiche Naturali

## Esercizio I: Ricerca di un elemento

**Sia L una lista definita da**

```
struct lista {int val; struct lista *next;} L;
```

**Scrivere in linguaggio C una funzione ricorsiva che preso in input L e un intero el, verifichi se esiste una occorrenza di el nella lista**

**Idea per la soluzione dell'esercizio:**

- Scorrere in avanti la lista in chiamate ricorsive e
  1. Se L=NULL, ritornare 0
  2. se L->val=el, ritornare 1
  3. altrimenti, ritornare il risultato della chiamata ricorsiva su L->next

back ✖ next

Federica 27/09/2007 5  Facoltà di Scienze  
Matematiche  
Fisiche Naturali

## Esercizio II: Rimozione di un elemento

Sia  $L$  una lista definita da

```
struct lista {int val; struct lista *next;} L;
```

Scrivere in linguaggio C una funzione ricorsiva che preso in input  $L$  e un intero  $el$ , rimuova una ricorrenza di  $el$  dalla lista (se ne esiste una).

**Idea per la soluzione dell'esercizio:**

- Scorrere in avanti la lista in chiamate ricorsive e
  1. se  $L=NULL$ , ritornare  $L$
  2. se  $L \rightarrow val=el$ , settare  $L=L \rightarrow next$  e ritornare  $L$
  3. altrimenti, associare a  $L \rightarrow next$  il risultato della chiamata ricorsiva su  $L \rightarrow next$  e al ritorno dalla chiamata ricorsiva restituire  $L$ .

back  next

Federica 27/09/2007 6  Facoltà di Scienze  
Matematiche  
Fisiche Naturali

## Esercizio III: Aggiunta di un elemento

Sia  $L$  una lista definita da

```
struct lista {int val; struct lista *next;} L;
```

Scrivere in linguaggio C una funzione ricorsiva che preso in input  $L$ , la modifichi in modo tale che ogni numero dispari sia seguito dal suo successore pari.

**Idea per la soluzione dell'esercizio:**

1. Dapprima scorrere la lista fino alla coda conchiamate ricorsive
2. Condizione di uscita: Se  $L=NULL$ , ritornare  $L$
3. Al ritorno dalle chiamate ricorsive:
  1. Se  $L \rightarrow val$  è dispari aggiungere un nuovo nodo tra  $L$  e  $L \rightarrow next$ .  
Attenzione qui a sistemare tutti i collegamenti!!!
  2. Ritornare la lista  $L$ .

back  next

The slide is titled "Esercizio completo (da consegnare)" in green. It contains a paragraph in green text describing the task: implementing a menu in C for list management using recursive functions. Below this is a bulleted list of operations. A red-bordered box highlights the submission deadline. At the bottom, there are navigation buttons: "back", a close button (X), and "next".

**Esercizio completo (da consegnare)**

**Si implementi in linguaggio C un menù di scelta multipla per la gestione "con funzioni ricorsive" di tutte le operazioni viste in precedenza. Dunque il menù permettere le seguenti operazioni:**

- Creazione di una lista a puntatori (con il comando **struct**) e successivo riempimento della lista con n elementi dati in input
- Stampa della lista
- Ricerca di un elemento,
- inserimento (in coda o dopo un elemento specifico)
- rimozione della prima occorrenza di un elemento dato in input
- Ricerca e stampa del massimo e del minimo della lista

**L'esercizio completo va consegnato via mail al tutor entro 5 giorni lavorativi**

back X next

This document was created with Win2PDF available at <http://www.win2pdf.com>.  
The unregistered version of Win2PDF is for evaluation or non-commercial use only.  
This page will not be added after purchasing Win2PDF.