

# Gestione di una lista – I

17 marzo 2006



Fabrizio Ciacchi

Adesso abbiamo intenzione di presentare un programma che permetta di memorizzare e stampare una lista composta da un determinato numero di interi. Sebbene il programma possa sembrare semplice, in realtà la sua stesura, ed il ragionamento posto alla base di essa, è particolarmente accattivante, anche perché non si possono commettere errori, ne sintattici, ne logici.

Innanzitutto dobbiamo partire con l'inclusione delle librerie, in questo caso includiamo solamente **stdio.h**, per le più comuni operazioni di input/output, e **malloc.h** per l'allocazione dinamica della memoria; nessuna libreria particolare deve essere inclusa, in quanto, come già detto prima, una lista è composta da elementi preesistenti nel linguaggio C.

```
#include <stdio.h>
#include <malloc.h>
```

Successivamente creiamo il tipo elemento, alla base della lista, che contiene un campo intero di nome "inf" ed un campo puntatore al tipo elemento di nome "pun":

```
/* struttura elementi della lista */
struct elemento {
    int inf;
    struct elemento *pun;
}
```

Si dovranno quindi creare i prototipi delle due funzioni adibite alla soluzione del nostro problema; la prima funzione ha il compito di creare la lista, chiedendo i dati di input all'utente tramite tastiera, che poi verrà restituita dalla funzione stessa; la seconda, invece, restituisce un void e prende in input la lista da stampare.

```
/* prototipi delle funzioni */
struct elemento *crea_lista();
void visualizza_lista(struct elemento *);
```

Il codice prosegue con l'ovvia creazione del main, la dichiarazione della una variabile lista di tipo puntatore ad elemento, l'esecuzione della funzione **crea\_lista()**, che ha il compito di "riempire" di valori la lista, e l'esecuzione della funzione **visualizza\_lista()** che stamperà a video tutti gli elementi della lista;

```
int main()
{
    struct elemento *lista; // puntatore della lista
    lista = crea_lista(); // crea la lista
    visualizza_lista(lista); // stampa la lista
}
```

Procediamo con la definizione del corpo della funzione **crea\_lista()**, la quale crea due puntatori ad elemento, uno di nome **p** e l'altro di nome **punt**; queste due variabili serviranno per scorrere la lista, infatti **p** è il puntatore al primo elemento della lista, mentre **punt** è un puntatore ausiliario che permette di scorrere la lista; la variabile **i** è l'indice del ciclo, mentre **n** serve a memorizzare il numero degli elementi che si intende inserire.

```
struct elemento *crea_lista()
{
    struct elemento *p, *punt;
    int i, n;
```

La variabile `n` viene inserita tramite tastiera dall'utente,

```
printf("n Specificare il numero di elementi... ");
scanf("%d", & n);
```

Se `n` vale 0, viene richiesto di creare una lista vuota, quindi si assegna a `p` il valore null,

```
if(n==0)
{
    p = NULL; // lista vuota
```

altrimenti si costruisce il primo elemento della lista, chiedendo il suo valore da tastiera:

```
} else {
    /* creazione primo elemento */
    p = (struct elemento *)malloc(sizeof(struct elemento));
    printf("nInserisci il primo valore: ");
    scanf("%d", & p->inf);
    punt = p;
```

GESTIONE DI UNA LISTA – II ➤

1 ... 33 34 35 ... 47

TUTTE LE LEZIONI