

Liste

Liste	<p>La lista è una struttura dati unidimensionale, sequenziale e dinamica. E anche una struttura omogenea, ma a differenza dell'insieme, ammette ripetizioni di elementi.</p> <p>Si può accedere direttamente solo al primo elemento della lista, mentre per accedere al generico elemento si deve scandire sequenzialmente gli elementi che lo precedono.</p> <p>La lunghezza è data dal numero di elementi al suo interno, e non dagli elementi unici.</p> <p>Si può ottenere una sottolista selezionando un intervallo; la lista vuota è sottolista di qualsiasi altra lista.</p>
Rappresentazioni: Vettore	<p>Una lista può essere realizzata usando un array, utilizzando una variabile <i>first</i> per indicare l'elemento in prima posizione nell'array ed una variabile <i>size</i> per indicare il numero di elementi in esso.</p> <p>Le problematiche di questa rappresentazione sono legate ai limiti della struttura vettore:</p> <ul style="list-style-type: none">- aggiungere un elemento in coda è semplice, mentre aggiungere un elemento in testa o eliminare un elemento comporta lo spostamento di tutti gli altri elementi;- se il vettore non è dinamico, la dimensione è fissa ed è determinata alla creazione della lista.
Rappresentazioni: Rappresentazione Collegata	<p>Nella rappresentazione collegata gli elementi sono dei nodi che contengono due informazioni: il valore che si vuole memorizzare nel "nodo" ed un puntatore al "nodo" successivo.</p> <p>Questa rappresentazione supera le limitazioni di quella con vettore.</p> <p>Una variante di questa rappresentazione è quella con doppi puntatori (o simmetrica) dove ogni elemento, oltre ad un puntatore al successivo, ha anche un puntatore all'elemento precedente.</p>
Rappresentazioni: Rappresentazione con Cursori	<p>La rappresentazione con cursori può essere utilizzata in linguaggi che non permettono la gestione dei puntatori; si va a simulare il funzionamento dei puntatori attraverso dei vettori che contengono rispettivamente gli elementi e la posizione nell'array del successivo, oltre a variabili che indicano inizio, fine e posizioni libere.</p> <p>Questa rappresentazione tende ad avere le limitazioni tipiche dei vettori, oltre ad essere difficile da gestire.</p>

Specifica Semantica Lista

TIPI:

- **Lista**: sequenza $\langle a_1, a_2, \dots, a_n \rangle$ di elementi di tipo tipoelem dove l' i -esimo elemento ha valore a_i e posizione $\text{pos}(i)$
- **Boolean**: insieme dei valori di verità

OPERATORI

creaLista() = L'

POST: $L' = \langle \rangle$

listaVuota(L) = b

POST: $b = \text{true}$ if $L == \langle \rangle$
 $b = \text{false}$ altrimenti

leggiLista(p,L) = a

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n$
POST: $a = a(i)$

scriviLista(a,p,L) = L'

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n$
POST: $L' = \langle a_1, a_2, \dots, a_{i-1}, a, a_{i+1}, \dots, a_n \rangle$

primoLista(L) = p

POST: $p = \text{pos}(1)$

fineLista(p,L) = b

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n+1$
POST: $b = \text{true}$ if $p == n+1$
 $b = \text{false}$ altrimenti

succLista(p,L) = q

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n$
POST: $q = \text{pos}(i+1)$

predLista(p,L) = q

PRE: $p = \text{pos}(i) \ 2 \leq i \leq n$
POST: $q = \text{pos}(i-1)$

insLista(a,p,L) = L'

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n+1$
POST: $L' = \langle a_1, a_2, \dots, a_{i-1}, a, a_i, a_{i+1}, \dots, a_n \rangle$ if $1 \leq i \leq n$
 $L' = \langle a_1, a_2, \dots, a_n, a \rangle$ if $i == n+1$
 $L' = \langle a \rangle$ if $i == 1$ and $L = \langle \rangle$

cancLista(p,L) = L

PRE: $p = \text{pos}(i) \ 1 \leq i \leq n$

POST: $L' = \langle a_1, a_2, \dots, a_{i-1}, a_{i+1}, \dots, a_n \rangle$

Strutture Lineari

Strutture Lineari	<p>Le strutture lineari sono strutture dati unidimensionali e sequenziali; l'ordine è rilevante e risulta utile a ritrovare e accedere agli oggetti al suo interno.</p> <p>Le strutture lineari si differenziano, tra le varie caratteristiche, in base alla modalità di accesso:</p> <ul style="list-style-type: none">- Accesso Diretto Consiste nell'accesso tramite indice, come nei vettori.- Accesso per Scansione Accesso all'elemento generico per scansione, come nelle liste.- Accesso agli Estremi Accesso come nelle code e pile. <p>Un'ulteriore distinzione è possibile in base al modo di agire dopo aver individuato la posizione:</p> <ul style="list-style-type: none">- Lettura del Valore in una Posizione- Cambio del Valore- Inserimento di un Nuovo Componente (Scrittura)- Rimozione di un Componente (Cancellazione)
--------------------------	--