

# Linked List

Created by Gianluca DeLucia  
N86001826)  
[gian.delucia@studenti.unina.it](mailto:gian.delucia@studenti.unina.it)

## **SCOPO:**

Utilizzare le linked list.

## **SPECIFICHE:**

```
void list ();  
pNodo crea_nodo(int elem);  
pNodo inserimento_lista_ordinata_ricorsivo(int elem, pNodo head);  
pNodo inserimento_lista_ordinata_iterativo (int elem, pNodo pn);  
pNodo cancella_nodo(int elem, pNodo handler);  
void stampa(pNodo handler);
```

## **DESCRIZIONE:**

### **a) Background del problema:**

inserire un elemento nella lista:

- 1) Metodo Ricorsivo
- 2) Metodo Iterativo

### **b) Algoritmo:**

```
function inser_lista_ricorsivo (in: elem, pn)  
  var pn, nuovo_nodo : nodo  
  var elem : integer  
  begin  
    if (pn == NULL OR pn->info >= elem)then  
      nuovo_nodo := crea_nodo(elem);  
      nuovo_nodo->link := pn;  
    endif  
    else if ( pn->info<= elem)then  
      nuovo_nodo := crea_nodo(elem);  
      pn->link := nuovo_nodo;  
    endif  
    else inser_lista_ordinata_ricorsivo(elem, pn->link);  
    endif  
  end  
end inserimento_lista_ricorsivo
```

```

function inser_lista_ordinata_iterativo (in: elem, pn)
    var nuovo_nodo, temp, precedente : nodo
    begin

        nuovo_nodo := crea_nodo (elem);
        temp := pn;
        precedente := NULL;

        while (temp != NULL AND temp->info <= elem)do
            precedente := temp;
            temp := temp->link;
        endwhile
        if (precedente != NULL)then
            precedente->link := nuovo_nodo;
            nuovo_nodo->link := temp;
        endif
        else
            nuovo_nodo->link = pn;
            pn := nuovo_nodo;
        endif
    end
end inser_lista_ordinata_iterativo

```

#### **RIFERIMENTI BIBLIOGRAFICI:**

<http://www.federica.unina.it/corsi/programmazione-ii-laboratorio/>  
 A. Murli, G. Laccetti, Laboratorio di Programmazione I, Ed. Liguori.  
 Criscuolo, F.P. Tramontano, Introduzione alla Programmazione, Algoritmi  
 Imperativi in C++, Ed. Manna.  
 G. Dromey, Algoritmi fondamentali, Ed. Jackson.  
 B. Kernighan, D. Ritchie, Linguaggio C, Ed. Jackson.  
 H. M. Deitel, P. J. Deitel, Corso completo di programmazione, Ed. Apogeo.

#### **LISTA DEI PARAMETRI:**

```

    int elem;
    int scelta, scelta2;
    char risp;
    nodo* head;

    typedef struct Nodo{
        int info;
        struct Nodo* link;
    }nodo;

```

#### **INDICATORI DI ERRORI:**

Nessuno.

#### **PROCEDURE AUSILIARIE:**

Nessuna.

#### **RACCOMANDAZIONI SULL'USO:**

Non utilizzare numeri che abbiano più di otto cifre.

**ESEMPIO D'USO:**

**a) Esempio di programma chiamante:**

```
void list ()
{
    int elem;
    int scelta, scelta2;
    char risp;
    nodo* head;

    head = NULL;

    do{

        cout<<"\t\t\t\tLinked List\n\n\n";

        cout<<"\n\nScegli cosa fare: \n\n";
        cout<<"\t1) Iserisci elemento\n\t2) Stampa lista\n\t3) Elimina\n\n";

        scelta = _getch();

        switch (scelta)
        {
            case '1' :
            {
                cout<<"\n\nScegliere un tipo di algoritmo da
                eseguire: \n\n";

                cout<<"\t1) Ricorivo\n\t2) Iterativo\n\n";
                scelta = _getch();

                switch (scelta)
                {
                    case '1' :
                    {
                        cout<<"\nNuovo elemento da
                        inserire ==>> ";

                        cin>>elem;
                        head =
                        inserimento_lista_ordinata_ricorsivo(elem, head);
                    }break;
                    case '2' :
                    {
                        cout<<"\nNuovo elemento da
                        inserire ==>> ";

                        cin>>elem;
                        head =
                        inserimento_lista_ordinata_iterativo(elem, head);
                    }break;
                }
            }break;
            case '2' :
            {
                if (head == NULL)
                {
                    cout<<"La lista e' vuota!";
                }
                else
            }
```

```

        {
            cout<<" Lista Ordinata : ";
            stampa(head);
        }

        }break;
    case '3' :
    {
        cout<<"\nElemento da eliminare ===>> ";
        cin>>elem;
        head = cancella_nodo(elem, head);
        cout<<"\nElemento eliminato con successo!";

        }break;
    }

    cout<<"\n\nVuoi tornare indietro? <s/n> ? ";
    cin>>risp;
    system ("cls");

}while((risp == 's') || (risp == 'S'));
}

```

#### ***b) Esempio di esecuzione:***

Scegli cosa fare:

- 1) Iserisci elemento
- 2) Stampa lista
- 3) Elimina

Scegliere un tipo di algoritmo da eseguire:

- 1) Ricorivo
- 2) Iterativo

Nuovo elemento da inserire ===>> 3

Lista Ordinata : 3