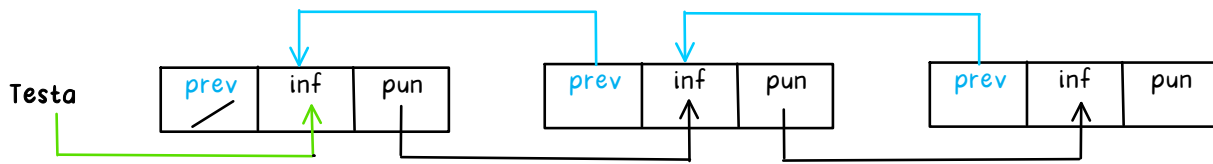


## 2.1-Liste Doppie

- Lista **doppiamente concatenata** o **doppia**, ciascun elemento contiene due puntatori:
  - un puntatore al **prossimo elemento**
  - un puntatore all'**elemento precedente**
  - Con le liste doppie è possibile **scorrere** la lista degli elementi **nelle due direzioni**



Nodo Lista Doppia	<pre>struct elem {      int inf;     elem* pun;     elem* prev; }  typedef elem* lista;</pre>
-------------------	---

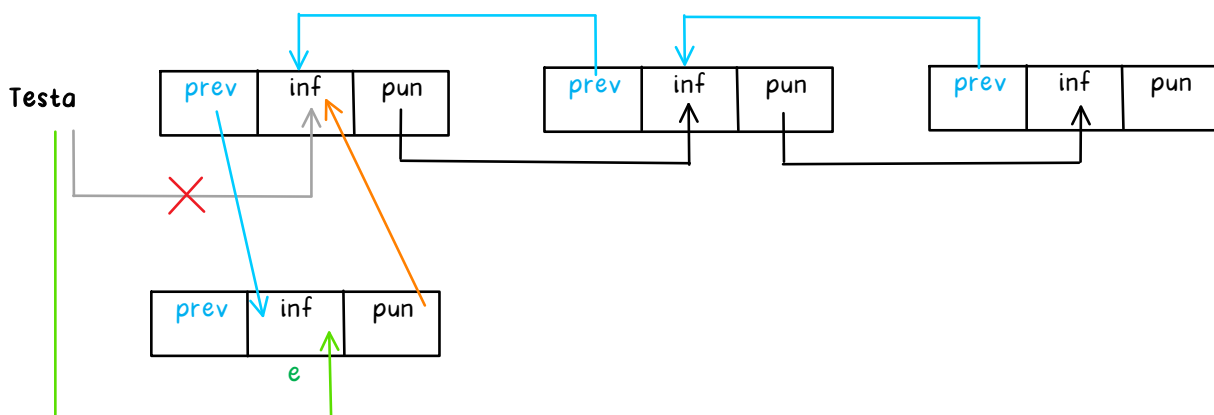
NB. Le primitive che cambiano da liste semplici a doppie sono:

- insertl\_elem
- delete\_elem

### Primitiva INSERT\_ELEM

-Funzione che aggiunge un elemento in testa alla lista e lo ritorna

```
lista insert_elem (lista l, elem* e) {  
  
    e->pun = l;  
    if (l != NULL)  
        l->prev = e;  
    e->prev = NULL;  
    return e;  
}
```



## Primitiva DELETE\_ELEM

-Funzione che cancella un elemento della lista

```
lista delete_elem (lista l, elem* e) {  
    if(l == e)  
        l = tail(l);  
    else  
        (e->prev)->pun = e->pun;  
  
    if(e->pun != NULL)  
        (e->pun) -> prev = e -> prev;  
  
    delete e;  
    return l;  
}
```

