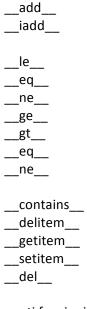
A) Sviluppare la classe **MyList** che implementa una sequenza la cui rappresentazione interna è basata sulla lista doppiamente collegata (*Double Linked List*). La classe deve supportate tutti i metodi della classe **list** di Python. In particolare deve supportare i metodi:

```
append(x)
extend(iterable)
insert(i, x)
remove(x)
pop([i])
clear()
index(x[, start[, end]])
count(x)
sort(key=None, reverse=False)
reverse()
copy()
```

Le parentesi quadre intorno ad un argomento nella firma di una funzione indicano che il parametro è opzionale. La semantica dei metodi deve essere la stessa di quella della classe **list**. I dettagli sono disponibili nel tutorial ufficiale di Python:

(https://docs.python.org/3.6/tutorial/datastructures.html).

La classe sviluppata deve supportare i seguenti operatori:



e le seguenti funzioni *built-in*:

len str bool

NOTA: L'implementazione fornita non può utilizzare funzioni e/o classi Python *built-in*. Si possono usare le funzioni definite nel modulo util.py disponibile sul sito del corso.

- **B)** Scrivere uno script verifica.py che testi tutte le funzionalità implementate. In particolare se l_uno è una lista e l_due = MyList(l_uno), allora la stessa operazione su entrambe le sequenze deve produrre lo stesso risultato.
- **C)** Definire una funzione Python che avendo come parametro input una sequenza di tipo **MyList** restituisca tutti i possibili suffissi di tale sequenza (es. $[a,b,c] \rightarrow []$, [c], [b,c], [a,b,c]). Definire una versione iterativa e una ricorsiva della funzione.