1. Scrivere una classe di base ClsBase in cui c'e` un metodo addAttr che controlla se la classe ha l'attributo di nome s e se tale attributo non e` presente allora aggiunge l'attributo s con valore v; in caso contrario non fa niente. Il metodo deve funzionare anche per le eventuali sottoclassi agendo sulla sottoclasse senza bisogno pero` di essere ridefinito nella sottoclasse.

2. Scrivere un decorator factory che prende on in input un tipo t genera un decoratore di classe che dota la classe di un metodo per contare il numero di variabili di istanza di tipo t dell'istanza per cui è invocato.

3. Scrivere nel file esercizio1.py una funzione generatrice generatore(L,X) che prende in input una lista L di numeri e una lista di interi X. Indichiamo con L_i l'i-esimo elemento di L e con X_i è l'i-esimo elemento di X (ovvero gli elementi di indice i-1 delle due liste rispettivamente). La funzione deve restituire un iteratore per il quale l'i-esima invocazione di next (con i<=lunghezza(X)), si comporta come segue:
   - se 0 <= x_i < lunghezza(L), next restituisce il valore ottenuto sommando i primi x_i valori di L_i
   - se x_i <0 oppure x_i >=lunghezza(L), next restituisce l'eccezione IndexError.