

**ALGORITMI E STRUTTURE DATI**  
**STATISTICA PER I BIG DATA**  
**APPELLO DEL 3 FEBBRAIO 2022**

In classe abbiamo studiato algoritmi di backtrack per il problema del target. Abbiamo una lista  $L$  di  $N$  interi positivi ed un target  $T$  e vogliamo verificare se esiste un sottoinsieme degli elementi della lista la cui somma è uguale ad  $N$ . In questo esame chiediamo un algoritmo che risolve il problema con il vincolo ulteriore che non è possibile raggiungere il target con un sottoinsieme che include due elementi che si trovano in posizione consecutive di  $L$ . Ad esempio, si consideri  $L = [2, 5, 8, 3, 9, 6]$ .

- (1) per il target  $T = 10$ , abbiamo  $L[0] + L[5] = 2 + 6 = 10$ .
- (2) per il target  $T = 7$ , abbiamo  $L[0] + L[1] = 2 + 5 = 7$ . La soluzione però non è ammissibile perché il target è ottenuto usando due elementi consecutivi di  $L$ .

**Istruzioni per la consegna.** Il codice da sviluppare consiste di una classe **TargetNC** derivata dalla classe **Target** che risolve il problema. Tutto il codice consegnato deve essere contenuto in un file con estensione **.py** che ha come nome il cognome dello studente scritto in minuscolo. Se il cognome contiene un apostrofo, uno spazio o un accento questi dovranno essere omessi.

La cartella che ha ricevuto contiene il pdf di questa traccia e il file **driver.py** che può essere usato per verificare il funzionamento della funzione progettata e i file **back.py** e **target.py**.