Seconda prova intermedia - 20/12/2022 Cognome, Matricola _____

La consultazione di dispense e dispositivi elettronici durante la prova è vietata. La valutazione dell'esame terrà conto solamente delle soluzioni riportate su questi fogli.

1. Si consideri un array $\boldsymbol{A}$ di elementi $a_0, \ldots, a_{n-1}$ positivi, e la formula

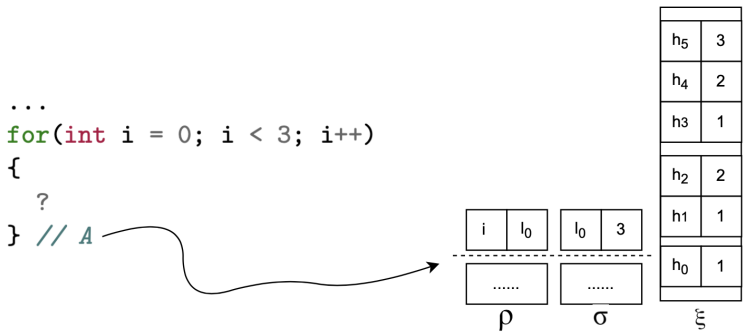
$$\sum_{i=0}^{k} a_i \leq \omega < \sum_{i=0}^{k+1} a_i \tag{1}$$

dove $\omega > 0$ e $k < n - 1$.

A. Si scriva una funzione `estrattore` che, partendo da $\boldsymbol{A}$ e ω
   - ordini $\boldsymbol{A}$, con un metodo a scelta dello studente;
   - determini, dall'array ordinato, se esiste k che soddisfa l'equazione (1);
   - se esiste tale k, allora restituisca un nuovo array $\boldsymbol{A}_k$ contenente solamente i k elementi ordinati $a_0, \ldots, a_k$;
   - se non esiste tale k, allora restituisca l'array $\boldsymbol{A}$ ordinato.

B. Dato $\boldsymbol{A} = [1 \mid 3 \mid 8 \mid 9]$, si diano:
   - un esempio di ω per cui tale k esiste;
   - un esempio di ω per cui tale k non esiste

2. Si consideri la seguente porzione di codice e la memoria qui riportata come rappresentazione del programma al punto `A`, precisamente prima dell'eliminazione dell'ultimo frame del ciclo `for`.



Si proponga un modo di complementare il codice al punto `?` al fine di creare esattamente la memoria rappresentata qui sopra.

3. Si consideri la definizione di liste linkate vista a lezione, e si risolvano le seguenti operazioni:

   A. la lista deve poter memorizzare, invece di singoli interi (`int info`), un array di valori in ciascun elemento, con una dimensione variabile per ogni elemento della lista;

   B. si crei una funzione `initialise` che:

      - prenda in input $n \geq 1$ e $z \geq 1$;
      - crei una lista di n elementi, dove l'elemento x-esimo contiene l'array dei primi x numeri maggiori strettamente di z.

      Per esempio, per $z = 10$ ed $n = 3$ vale

      ```
      initialise(3, 10) = [11] ---> [11|12] ---> [11| 12| 13]
      ```

   C. si disegni la memoria vera dopo aver eseguito `initialise(3, 10)` in un vostro programma.

4. Si fornisca, usando i principi della programmazione ad oggetti visti a lezione, un programma Python che permetta di ordinare una lista di oggetti qualunque.

   Suggerimento: si consiglia di implementare una funzione `ordina` che prende una lista di *oggetti*; la funzione deve quindi restituire la lista di oggetti ordinati, usando la vostra strategia preferita. L'unica cosa a cui dovete pensare è come "confrontare" due oggetti che non conoscete a priori. *Potete "uniformare" gli oggetti della lista in modo da assumere di poterli confrontare?*