

## ESERCIZIO 0 - PARTE A

### - PILE E CODE DI STRINGHE CON RAPPRESENTAZIONE VETTORIALE -

F. MOGAVERO

Implementare due librerie di funzioni in **Linguaggio C** per la gestione di strutture dati dinamiche di tipo **pila** e **coda** i cui dati siano **stringhe alfanumeriche**. Ogni struttura dovrà essere implementata concretamente sfruttando **vettori di dimensione variabile** con efficiente **ridimensionamento automatico** dello spazio allocato.

Le funzionalità da realizzare sono di seguito elencate:

- (1) costruzione e distruzione della struttura dati;
- (2) operazioni elementari (**inserimento** di un dato elemento; **rimozione**, **rimozione con lettura** e **lettura non distruttiva** dell'elemento immediatamente accessibile; **controllo di vuotezza** della struttura; **dimensione** della struttura);
- (3) operazioni non-elementari (**clonazione** della struttura; **controllo di uguaglianza** di due strutture; **controllo di esistenza** di un dato elemento nella struttura; operazione di **applicazione di una funzione** agli elementi della struttura: **funzione map**; operazione di **accumulazione di un valore** estratto dagli elementi della struttura: **funzione fold**).

Al fine di poter testare adeguatamente le librerie sopra descritte, si richiede di definire (esternamente alle stesse, in un opportuno file di test richiamato dal “main”) un insieme di procedure che implementi le seguenti funzionalità:

- (1) **scelta della struttura** (pila vs coda);
- (2) **popolamento della struttura** precedentemente scelta con  $n$  stringhe generate casualmente, dove  $n$  è un parametro dato dall'utente in ingresso;
- (3) **visualizzazione di tutte stringhe** contenute nella struttura (effettuata per mezzo della **funzione map**);
- (4) **concatenazione** (effettuata per mezzo della **funzione fold**) di tutte le stringhe con lunghezza minore o uguale a  $n$  presenti nella struttura e relativa visualizzazione del risultato, dove  $n$  è un parametro dato dall'utente in ingresso;
- (5) **applicazione delle funzioni “uppercase” e “lowercase”** a tutte le stringhe contenute nella struttura.

Inoltre, da un opportuno menu, dovrà essere possibile operare sulla struttura scelta attraverso le funzioni di libreria di cui al punto (2).

Il codice sorgente prodotto dovrà seguire pedissequamente (sia nei nomi delle funzioni di libreria, sia nella strutturazione e nei nomi delle diverse directory e file “.c” e “.h”) la forma riportata nel template Exercise0A.zip associato al presente esercizio.