

NOTA Gli esercizi **1, 2, 3 e 4** si riferiscono alla sessione di Laboratorio #3; gli esercizi **5, 6 e 7** alla sessione #4. Gli studenti devono presentarsi alla sessione #4 con i programmi della sessione #3 scritti e funzionanti.

### Esercizio 1

Definire un parametro (tramite una direttiva `#define` o una dichiarazione `const`) chiamato CAPIENZA avente valore 10. Definire un tipo di dato, chiamato TipoL, corrispondente ad una lista sequenziale in grado di contenere al massimo CAPIENZA dati di tipo carattere.

Scrivere un programma C che legge da tastiera CAPIENZA *lettere minuscole tutte diverse tra loro* e le memorizza una dopo l'altra in una variabile (chiamata Lista) di tipo TipoL. Assicurarsi che i caratteri inseriti siano effettivamente lettere minuscole e che essi siano diversi tra di loro è a cura dell'utente del programma.

Una volta conclusa la fase di input, il programma deve stampare a schermo in sequenza i caratteri contenuti in Lista.

### Esercizio 2

Modificare il blocco `main` in modo che l'utente possa inserire nella lista sequenziale un numero di caratteri a sua scelta, purché non superiore a CAPIENZA. La fase di input deve terminare quando l'utente immette uno speciale carattere TERMINATORE definito con una `#define` (tale carattere non va inserito nella lista), oppure quando sono state inserite nella lista CAPIENZA lettere minuscole.

Se l'utente inserisce un carattere che non appartiene all'alfabeto minuscolo ed è diverso dal TERMINATORE, il carattere non viene inserito nella lista sequenziale e viene stampato il messaggio *“Il carattere inserito non è valido ed è stato ignorato”*.

### Esercizio 3

Produrre una versione modificata del programma dell'Esercizio 2 che, ogniqualvolta l'utente immette una lettera minuscola, verifica che essa sia effettivamente diversa da tutte quelle già contenute in Lista. Nel caso in cui la lettera immessa dall'utente risulti uguale ad una di quelle già presenti in Lista, essa non viene inserita nella lista sequenziale e viene stampato il messaggio d'errore *“Il carattere inserito è un doppione ed è stato ignorato.”*.

## Esercizio 4

Nel programma dell'Esercizio 3, spostare il codice che esegue la stampa del contenuto della lista sequenziale `Lista` all'interno di una funzione (chiamata **Stampa**) che ha come parametro d'ingresso un dato di tipo `TipoL`.

Modificare il blocco `main` in modo da usare la funzione `Stampa` per stampare il contenuto della lista sequenziale.

## Esercizio 5

Scrivere una funzione (chiamata **Minimo**) che ha come parametro d'ingresso un dato di tipo `TipoL` e restituisce il minimo carattere tra quelli contenuti nella lista sequenziale ricevuta in ingresso, ovvero quello più vicino all'inizio dell'alfabeto minuscolo. E' richiesto che `Minimo` operi correttamente solo se la lista che riceve in ingresso contiene almeno un carattere.

Nel programma dell'Esercizio 4, aggiungere al blocco `main` una parte che verifica se `Lista` contiene almeno 1 elemento e, in tal caso, chiama la funzione `Minimo` (passandole `Lista`) e stampa a schermo il carattere restituito da `Minimo`.

NOTA: in C l'insieme dei `char` è *ordinato*, dunque è possibile fare confronti del tipo `if (Char1 < Char2)` tra dati di tipo `char`.

## Esercizio 6

Nel programma dell'Esercizio 5, spostare il codice che esegue la fase di input all'interno di una funzione (chiamata **EseguInput**) che ha come parametro d'ingresso un dato di tipo `TipoL`. Modificare il `main` in modo da usare `EseguInput`.

## Esercizio 7

Modificare la funzione `Minimo` in modo che riceva anche un secondo parametro di ingresso di tipo `char`, chiamato `Soglia`. La funzione modificata restituisce il minimo carattere di valore superiore a Soglia tra quelli contenuti nel dato di tipo `TipoL` che riceve in ingresso. Se la lista non contiene alcun carattere superiore a `Soglia`, la funzione restituisce `TERMINATORE`. Si modifichi la chiamata a `Minimo` presente nel blocco `main` in modo da usare la nuova versione della funzione.

Scrivere una funzione (chiamata **Ordina**) che riceve in ingresso un dato di tipo `TipoL` e ne modifica il contenuto in modo che i caratteri che esso contiene risultino in ordine di valore crescente. **Il codice di `Ordina` deve usare la funzione `Minimo`.**

Modificare il blocco `main` in modo che, se `Lista` contiene almeno un elemento, dopo aver stampato il carattere di minimo valore, prima usi `Ordina` per ordinare il contenuto della lista sequenziale, poi stampi il contenuto della lista modificata.