

1. Sviluppare una function C che, dato come parametro di input un array di tipo **struct punto [double x; double y]** e il suo size, determina e restituisce come parametro di output la massima distanza tra i punti.
2. Sviluppare una function C che, dati come parametri di input un array di tipo **struct punto [double x; double y]** e il suo size, determina e restituisce come parametri di output gli indici dei due punti che hanno distanza minima.
3. Sviluppare una function C che effettua la ricerca sequenziale di una chiave di tipo stringa in un elenco di stringhe (rappresentato mediante un array di puntatori a **char**). La function deve usare la function C **strcmp**. La function restituisce come parametri di output un dato logico e inoltre, se la ricerca ha successo restituisce anche l'indice del primo elemento uguale alla chiave.
4. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametro di output il numero di parole di tre lettere. Nel testo le parole sono separate da un unico 'spazio'.
5. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametro di output il numero di parole che terminano in 'are'. Nel testo le parole sono separate da un unico 'spazio'.
6. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametro di output il numero di parole che iniziano con 'a' e terminano con 'e'. Nel testo le parole sono separate da un unico 'spazio'.
7. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametro di output il numero di parole che hanno almeno 5 vocali. Nel testo le parole sono separate da un unico 'spazio'.
8. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametri di output la parola di lunghezza massima e la sua lunghezza. Nel testo le parole sono separate da un unico 'spazio'.
9. Sviluppare una function C che, data come parametro di input una stringa che rappresenta un testo in italiano, determina e restituisce come parametri di output la parola di lunghezza massima e la posizione di inizio della parola nella stringa. Nel testo le parole sono separate da un unico 'spazio'.
10. Sviluppare una function C che, dati come parametri di input un array di **int** e il suo size, determina e restituisce come parametro di output il secondo più grande elemento dell'array.
11. Sviluppare una function C che, dati come parametri di input un array di **int**, il suo size e un **int k**, determina e restituisce come parametro di output il **k**-imo più grande elemento dell'array.
12. Sviluppare un main C che legge da tastiera un valore di un **int k** (compreso tra 5 20) e visualizza una tabellina in cui la prima riga contiene il testo
"n n^2 n^3 nlog(n) exp(n)"

e in ogni successiva riga appaiono i valori (tipo **double**) di n , di n^2 , di n^3 , di $n \cdot \log(n)$, di e^n , per tutti valori di n da 2 a k .

13. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametro di output l'array (di size 21) del numero delle occorrenze delle 21 lettere dell'alfabeto italiano.
14. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametro di output l'array (di size 21) del numero delle occorrenze dell'evento 'a' precede ognuna delle 21 lettere dell'alfabeto italiano (cioè il numero di volte in cui accade che 'a' precede 'a', il numero di volte in cui accade che 'a' precede 'b', il numero di volte in cui accade che 'a' precede 'c',.....
15. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametro di un dato logico che indica se il testo dell'array è un *pangramma*, ovvero è un testo che contiene, almeno una volta, tutte le 21 lettere dell'alfabeto italiano.
16. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametro di output il carattere più frequente.
17. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametro di output il carattere meno frequente.
18. Sviluppare una function C che, dati come parametri di input un array di **char** e il suo size, determina e restituisce come parametri di output il carattere più frequente e il carattere meno frequente.
19. Sviluppare una function C che, dati come parametri di input un array 2D di **int**, il numero delle righe, il numero delle colonne e un **int k**, determina e restituisce come parametro di output la somma degli elementi della **k**-sima riga.
20. Sviluppare una function C che, dati come parametri di input un array 2D di **int**, il numero delle righe, il numero delle colonne e un **int k**, determina e restituisce come parametro di output la somma degli elementi della **k**-sima colonna.
21. Sviluppare una function C che, dati come parametri di input un array 2D di **int**, il numero delle righe e il numero delle colonne, determina e restituisce come parametro di output il massimo tra le somme degli elementi di ogni riga.
22. Sviluppare una function C che, dati come parametri di input un array 2D di **double**, il numero delle righe e il numero delle colonne, determina e restituisce come parametro di output il massimo tra le somme degli elementi di ogni colonna.
23. Sviluppare una function C che, dati come parametri di input un array 2D di **float**, il numero delle righe, il numero delle colonne, un **int p** e un **int q** (con **p** minore di **q**), determina e restituisce come parametro di output la somma degli elementi che hanno gli indici di riga e di colonna compresi tra **p** e **q**.

24. Sviluppare una function C che, dati come parametri di input un array di **int** e il suo size, determina e restituisce come parametri di output l'indice di inizio e la lunghezza della più lunga sequenza ordinata (senso crescente) contenuta nell'array.
25. Sviluppare una function C che, dati come parametri di input **int n** e i **double z_iniz_1, z_iniz_2, alfa_iniz_1**, determina e restituisce come parametri di output l'**n**-simo elemento della successione (doppia)

$$z_i = \alpha_{i-1} + z_{i-2}$$

$$\alpha_i = \alpha_{i-1} + z_{i-1}$$

26. Sviluppare una function C che, dati come parametri di input un array di **char** (con elementi diversi dallo 'spazio') e il suo size, restituisce come parametro di output lo stesso array di input, in cui sono stati eliminati tutti gli elementi duplicati (gli elementi non duplicati devono essere "compattati", per cui il numero di elementi diversi dallo spazio in genere diminuisce; gli elementi duplicati sono sostituiti dallo 'spazio' e si trovano alla fine dell'array).
27. Sviluppare una function C (**moda**) che, dati come parametri di input un array di **int** e il suo size, determina e restituisce come parametro di output il valore più frequente (che in statistica prende il nome di *moda*)