

Nome: _____ Cognome: _____ Matr: _____ Postazione _____

Scrivere un programma in linguaggio C (chiamare il progetto con la propria <matricola>) che abbia il comportamento descritto nel seguito. Il tempo a disposizione è di **120 minuti**. Al termine del tempo gli elaborati salvati su **U:** verranno raccolti automaticamente dal sistema. Eventuali documenti sono disponibili in **T:\Bertozzi**, si consiglia di usare **wordpad** per leggere i file di testo.

Il file “parole.txt” contiene, riga per riga, coppie di parole separate da uno o più spazi
Realizzare un programma in C che:

1. Memorizza il contenuto del file “parole.txt” in una struttura dati, scelta a piacere, e allocata dinamicamente
2. Per ciascuna coppia di parole memorizzata invoca la funzione `int check_palindromi(??)` che, avvalendosi a sua volta di altre funzioni, controlla se nella coppia di parole passata vi siano parole palindroma (si definisce palindroma una parola che letta in senso inverso rimane comunque uguale, esempio “kayak”). Detta funzione restituisce il numero di parole palindroma presenti nella coppia, ovvero 0 se non ve ne sono, 1 se almeno una delle parole è palindroma o 2 se le sono entrambe.
Tale parte di codice, è valutata maggiormente se realizzata sfruttando la ricorsione.
3. Per ciascuna coppia di parole memorizzata invoca la funzione `int check_anagrammi(??)` che verifica se le due parole nella coppia sono una anagramma dell'altra ovvero se permutando le lettere della prima parola è possibile ottenere la seconda (o viceversa). Esempio: “senatrice” e “cestinare”. La funzione restituisce 1 se le due parole sono una anagramma dell'altra e 0 se non lo sono.
4. A valle delle invocazioni delle funzioni la `main()` stampa opportuno messaggio che indica se le parole sono palindroma e/o una anagramma dell'altra

Esempio di esecuzione

La coppia 'image' 'magie' non contiene parole palindroma e sono anagrammi una dell'altra

La coppia 'ingegni' 'onorarono' contiene 2 parole palindroma

La coppia 'palissandro' 'omonomo' contiene 1 parola palindroma

Il codice va sviluppato nell'ordine indicato. Per ogni punto implementato si verifichi il corretto funzionamento. La correzione termina al primo punto non implementato correttamente.