

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 di laboratorio. Eventuali documenti sono disponibili in T:\Bertozzi, si consiglia di usare **wordpad** per leggere i file di testo.

La “grafica della tartaruga” permette di disegnare sullo schermo impartendo dei comandi a una ipotetica tartaruga che si muove su di una griglia $r \times c$. La tartaruga ha una penna, che può essere sollevata o abbassata e di conseguenza scrivere o meno traccia del suo passaggio. I comandi che si possono dare alla tartaruga sono i seguenti: T – commuta la penna (abbassa la penna se era alzata o viceversa), A - avanza di una casella (salta nella cella immediatamente davanti), D - gira a destra (ruota di 90° in senso orario senza spostarsi).

A partire dallo scheletro di codice fornito, si sviluppino le seguenti parti di codice:

1. (10) la parte di codice che chiede all'utente il nome di un file ASCII che contiene nella prima riga le dimensioni della griglia (numero righe e numero colonne), nella seconda riga la posizione e lo stato iniziale della tartaruga (riga, colonna, orientazione e stato penna). Si scelga opportuna struttura dati per memorizzare lo stato della tartaruga e si allochi dinamicamente la griglia;
2. (2) una funzione **void stampa(??)** che stampa il contenuto della griglia utilizzando 'X' per le celle in cui è stato “scritto”, '.' per le celle “pulite”, "O" per la posizione della tartaruga con la penna alzata e "o" per la posizione della tartaruga con la penna abbassata;
3. (2) una funzione **void stampa_tartaruga(??)** che stampa lo stato della tartaruga;
4. (2) nel file, dopo le informazioni iniziali già discusse, è contenuta una stringa che contiene i vari comandi. Si implementi una funzione **char comando(??)** che restituisce il prossimo comando memorizzato nel file oppure '\0' se si arriva a fine file o comunque viene letto un comando non valido;
5. (2) la parte di codice che, sfruttando le funzioni definite nei punti successivi, esegua il comando letto al punto precedente invocando le funzioni descritte nel seguito;
6. (8) una funzione **void avanza(??)** che fa avanzare la tartaruga eventualmente aggiornando la griglia a seconda dello stato della penna. La tartaruga non può superare i bordi della griglia e, di conseguenza, ignorare eventuali avanzamenti nel caso;
7. (2) una funzione **void ruota(??)** che ruota di 90° in senso orario la tartaruga;
8. (2) una funzione **void toggle(??)** che commuta lo stato della penna della tartaruga ed eventualmente scrive nella griglia;

Il codice va sviluppato nell'ordine indicato e per ogni punto va adeguatamente verificato il funzionamento prima di passare al successivo.