

# Esame di Programmazione Cl. B (Parte 2)

26 Giugno 2024

Ingegneria e Scienze Informatiche A.A. 2023-2024

Per questo esame, devi implementare una versione semplificata del classico gioco della Battaglia Navale. L'esercizio prevede la creazione di funzioni per gestire una griglia di gioco 10x10, posizionare le navi, visualizzare la griglia e gestire gli attacchi. Saranno forniti i file `lib.h` con i prototipi delle funzioni e `main.c` per testare le funzioni che svilupperai. L'unico file che dovrai modificare e consegnare sarà `lib.c`.

## Punteggio Totale: 16 Punti

Il punteggio sarà ripartito equamente tra le funzioni richieste, con 4 punti assegnati per ciascuna funzione completata correttamente.

## Specifiche delle Funzioni

### 1. Inizializza Griglia (4 punti)

- **Obiettivo:** Preparare la griglia di gioco, inizializzandola con acqua in ogni cella.
- **Input:** Un puntatore a una griglia di gioco.
- **Output:** Nessuno (la griglia viene modificata in-place).
- **Implementazione:** La griglia sarà un array 2D (definito da una macro `DIM_GRIGLIA` pari a 10), dove ogni cella viene inizializzata a 'A' per acqua.

### 2. Posiziona Nave (4 punti)

- **Obiettivo:** Posizionare una nave sulla griglia.
- **Input:** Puntatore alla griglia, coordinate iniziali (x, y), dimensione della nave e direzione (VERTICALE o ORIZZONTALE).
- **Output:** Valore int (1 per successo, 0 per fallimento).
- **Implementazione:** Le coordinate iniziali (x, y) corrispondono all'estremo superiore della nave se posizionata verticalmente e all'estremo sinistro se posizionata orizzontalmente. La funzione controlla se la nave può essere posizionata senza sovrapposizioni o uscire dai bordi della griglia. Se possibile, posiziona la nave (rappresentata da 'N').

### 3. Mostra Griglia (4 punti)

- **Obiettivo:** Stampare la griglia di gioco.
- **Input:** Puntatore alla griglia.

- **Output:** Stampa la griglia sulla console.
- **Implementazione:** La funzione stampa la griglia, mostrando 'A' per acqua, 'N' per nave, 'X' per colpito e 'M' per mancato.

#### 4. Attacco (4 punti)

- **Obiettivo:** Eseguire un attacco in una specifica posizione della griglia.
- **Input:** Puntatore alla griglia e le coordinate dell'attacco (x, y).
- **Output:** Risultato dell'attacco ('X' per colpito, 'M' per mancato, 'R' per ripetuto se già attaccato).
- **Implementazione:** Modifica la griglia basata sul risultato dell'attacco e ritorna il risultato dell'azione.

#### Istruzioni per la Consegna

- **File da Consegnare:** `lib.c` contenente tutte le funzioni implementate.
- **File Forniti:** `lib.h` con prototipi di funzioni e `main.c` per testare le funzioni implementate.
- **Tempo a disposizione:** 1 h 30'

#### Note

- Assicurati di testare adeguatamente il codice per garantire che ogni funzione operi correttamente prima della consegna.
- Commenta il codice ove necessario per spiegare parti complesse o scelte di implementazione.
- La chiarezza e la leggibilità del codice saranno valutate come parte dell'esame.

```

1 #ifndef LIB_H
2 #define LIB_H
3
4 #define DIM_GRIGLIA 10 // Dimensione della griglia di gioco, 10x10
5 #define VERTICALE 1
6 #define ORIZZONTALE 0
7
8 // Dichiarazione delle funzioni
9 void inizializzaGriglia(char griglia[DIM_GRIGLIA][DIM_GRIGLIA]);
10 int posizionaNave(char griglia[DIM_GRIGLIA][DIM_GRIGLIA], int x, int y, int dimensione, int
    direzione);
11 void mostraGriglia(char griglia[DIM_GRIGLIA][DIM_GRIGLIA]);
12 char attacco(char griglia[DIM_GRIGLIA][DIM_GRIGLIA], int x, int y);
13 #endif

```