

Domino 2D

Habtoune Hicham matricola 902308

Sommario

Introduzione	2
Struttura del codice	2
Modalità interattiva	4
Modalità AI	4

Introduzione

Il progetto riguarda l'implementazione del gioco del domino in 2D in linguaggio C compilato nel seguente modo:

```
gcc -O2 -std=c99 --pedantic *.c -o iap -lm
```

Le tessere possono essere posizionate sia in verticale che in orizzontale a seconda della preferenza o possibilità del giocatore.

Il numero di tessere con cui giocare viene chiesto all'utente, mentre i numeri delle tessere vengono generate casualmente, con numeri che vanno da 1 a 6.

Ogni partita include sempre le 3 tessere speciali che possono essere accostate a qualunque tessere:

- [0|0]: può essere accostata a qualunque altra tessera;
- [11|11]: somma 1 a tutte le cifre di tutte le tessere sul piano di gioco tranne il 6 che diventa 1;
- [12|21]: copia “a specchio” la tessera adiacente.

Il progetto include sia la modalità interattiva che quella ai.

Struttura del codice

Alberatura del progetto:

```
|— docs
    |— html
    |— home.html
|— source
    |— Doxyfile
    |— logic.c
    |— logic.h
    |— main.c
    |— Makefile
    |— tessere.c
    |— tessere.h
```

Il codice è suddiviso in tre file principali: ***main.c***, ***logic.c***, ***tessere.c***.

Sia ***logic.c*** che ***tessere.c*** hanno un proprio file header (***logic.h*** e ***tessere.h***), dove vengono specificate le loro firme e gli appositi commenti per la documentazione di Doxygen.

Inoltre il progetto dispone di un file Makefile per una compilazione del codice più intuitiva.

tessere.c: contiene le funzioni principali per la manipolazione delle tessere, nonché di tutte le altre strutture dati;

logic.c: contiene le funzioni per la logica del gioco e la gestione di una partita interattiva e ai.

La cartella docs contiene tutto ciò che riguarda la documentazione del progetto generata con Doxygen

Il codice è suddiviso in tre principali strutture dati: **Tessera**, **Row**, e **Board**.

- **Tessera**: Rappresenta una tessera del gioco con due numeri, un numero identificativo, e alcune informazioni aggiuntive come lo stato di selezione e l'orientamento verticale.
- **Row**: Rappresenta una riga della scacchiera di gioco, contenente un array di tessere. La struttura gestisce in modo dinamico la sua capacità e la dimensione effettiva dell'array, quindi ha una funzione di vettore.
- **Board**: Rappresenta la scacchiera di gioco, costituito da un array di righe (**Row**). Gestisce anche il punteggio della singola partita.

Modalità interattiva

Questa modalità è gestita dalla funzione ***game_start*** sul file ***logic.c***.

All'interno di un loop principale, il gioco continua fino a quando non è concluso o ci sono ancora tessere speciali disponibili. All'interno di questo loop, un altro loop gestisce la scelta della tessera da parte dell'utente. L'utente può inserire un numero positivo per selezionare una tessera normale o un numero negativo per selezionare una tessera speciale. La funzione verifica la validità dell'input, segna la tessera come selezionata e tenta di posizionarla sul tabellone. Se la posizione è valida, la tessera viene rimossa dall'array corrispondente.

Il punteggio del giocatore viene quindi aggiornato, e la visualizzazione del tabellone viene aggiornata. Questo processo continua finché il gioco non è concluso. Alla fine del gioco, la funzione stampa un messaggio indicando che non ci sono più tessere giocabili e successivamente libera la memoria.

Modalità AI

Questa modalità è gestita dalle funzioni ***game_start_ai*** e ***best_match*** sul file ***logic.c***.

Le funzioni servono a trovare la miglior partita partendo da ogni tessera tra quelle generate inizialmente, per ogni tessera viene costruita una scacchiera con un proprio punteggio inserendo la miglior tessera tra quelle possibilmente inseribili, se non si può inserire nessuna tessera viene inserita una tessera speciale. Quindi con N tessere, avrò N partite possibili. Nella modalità ai non sono state implementate le tessere verticali.