

Battaglia navale: Riepilogo funzioni

Jacopo Acquistapace, Giulio Cordova

April 27, 2019

1 Regole del gioco

Le regole del gioco sono descritte nel file regole.txt e qui sintetizzate:

- Il gioco prevede lo scontro fra due flotte di navi: la tua e quella del computer. Il numero delle navi e la dimensione della griglia sono le stesse per entrambi: di default si gioca su una griglia 8x8 con 1 nave da uno, 2 navi da due e 3 navi da tre a testa. A scelta si può decidere il numero di navi e la dimensione della griglia.
- Le navi si possono disporre in orizzontale o in verticale, ma non possono essere adiacenti in nessuna direzione (compresa la diagonale). Di default le navi vengono disposte casualmente dal calcolatore, ma si può scegliere di giocare inserendole manualmente. L'inserimento prevede di piazzare per prime le navi da tre, poi da due e infine da uno. Per posizionare una nave si decide la casella in cui posizionarle e poi la direzione (sud o est).
- E' possibile scegliere tra tre livelli di difficoltà:
 - La modalità facile prevede una generazione pseudo-casuale da parte del computer di coordinate su cui sparare. Le caselle già colpite non vengono prese in considerazione. In questa modalità la percentuale di caselle colpite sui tiri totali 20% (una nave colpita ogni 5).
 - La modalità intermedia prevede che il computer non spari sulle caselle adiacenti a navi già colpite. La percentuale colpite/totali è del 26% (una nave colpita ogni 4 tiri).
 - La modalità difficile dà la priorità a colpire intorno a pezzi di navi già colpiti ma non affondati. Nel caso in cui non ci fossero l'attacco funziona come per la modalità intermedia. La percentuale colpite/totali è del 30% (una nave colpita ogni 3 tiri). Le percentuali sono state calcolate su un campione di 100000 partite (i test sono avviabili col file 06test.c).
- E' possibile scegliere chi gioca per primo: a seguito di test si è osservato che a parità di livello di gioco, il giocatore che gioca per primo è avvantaggiato (vince in media il 53% delle partite, su un campione di 100000 partite). Si è introdotto perciò la scelta, che può essere manuale o casuale.
- Sullo schermo verranno visualizzate due griglie: nella prima a sinistra vi è la matrice delle navi del giocatore. Le navi non ancora colpite sono indicate con 1, 2 o 3, a seconda della dimensione. Quando vengono colpite, si assegna rispettivamente il carattere a, b, c sia che siano colpite in parte o completamente affondate. La griglia di destra indica la matrice del computer. Lì si possono vedere gli attacchi eseguiti fino a quel momento.
- Per colpire una casella si deve inserire una lettera maiuscola (coordinata riga) e un intero (coordinata colonna). Ad ogni turno si può effettuare un solo attacco, dopo di che il turno passa all'avversario
- Abbiamo condizioni periodiche di bordo, ovvero il primo elemento e l'ultimo di ogni riga/colonna sono adiacenti.
- Il gioco finisce quando tutte le navi sono completamente affondate.

2 Funzioni introdotte

Per implementare il gioco sono state introdotte diverse nuove funzioni: possiamo suddividerle grossolanamente in:

2.1 Funzioni di controllo

Sono funzioni che provvedono al corretto funzionamento delle altre, in particolare per l'input/output delle informazioni con l'utente, tra queste:

- `EMPTY_STDIN`: pulisce la stdin dopo che è stato letto un dato inserito dall'utente con la `scanf`
- `BUFFER`: verifica la corretta lettura e salvataggio dei dati forniti in input dall'utente ad opera della `scanf`
- `COND_BOARD`: realizza le condizioni al bordo sui parametri delle funzioni del gioco, in particolare in tutti i casi in cui il calcolatore controlla le caselle adiacenti ad altre.
- `VERIFY`, `VERIFY_SPEC`: controllano la fattibilità di attacchi o disposizioni di navi nelle caselle decise dall'utente o generate dal calcolatore

2.2 Funzioni per la preparazione del gioco

Sono le funzioni che preparano l'avvio del gioco creando le griglie di dimensioni opportune e realizzando la disposizione delle navi.

- `CREA_AREA`, `INIT`: creano ed inizializzano le griglie di gioco
- `PRINT_AREA_MIA`, `SCHEMA`: stampano le griglie in output, con accorgimenti diversi per quella dell'utente e quella con la quale l'utente attacca il computer.
- `PUT_MAN`, `PUT_ONE`, `PUT_TWO`, `PUT_THREE`, `PUT`: funzioni con le quali si inseriscono le navi nelle griglie di entrambi i giocatori. La funzione `PUT` è quella che gestisce l'interfaccia con l'utente, ed al suo interno chiama le `PUT_1,2,3` per le navi da 1,2,3 rispettivamente. Ognuna di queste utilizza `PUT_MAN` per inserire ogni nave.

2.3 Funzioni di modalità

Sono funzioni di interfaccia con l'utente che permettono di personalizzare il gioco e deciderne le modalità

- `INSER_NAVI`: chiede all'utente se inserire le navi manualmente o in modo automatico e quindi chiama la funzione `PUT`
- `CUSTOM`: chiede all'utente se giocare con le impostazioni predefinite o se personalizzare dimensioni della griglia, numero di navi ecc., verificando la realizzabilità delle specifiche richieste dall'utente.
- `DIFFI`: chiede all'utente di scegliere un livello di difficoltà.
- `PRIMO`: stabilisce chi gioca per primo (si può scegliere manualmente o casualmete)

2.4 Funzioni di gioco

Sono le funzioni che realizzano il gioco nei suoi turni e che si ripetono nella funzione `GIOCA` fino alla vittoria di uno dei due avversari.

- `ESEGUI_MOSSA`: esegue l'attacco, prendendo in input le coordinate e restituendo il risultato dell'attacco

- AFFONDATA_TRE: stesso scopo di ESEGUI_MOSSA per le navi di dim 3, separata in quanto più complessa
- ATTACCO_COMPUTER: genera le coordinate dell'attacco del computer in modo casuale (facile) oppure chiamando le funzioni INTERMEDIA e DIFFICILE per gli altri livelli di difficoltà.
- VITTORIA: Conteggia le navi presenti sulle griglie per stabilire il vincitore quando in una delle griglie tutte le navi sono state affondate
- GIOCA1: Funzione simil-gioca utilizzata per runnare test: stabilisce chi vince la partita o se viene generato un errore.
- GIOCA2: Funzione simil-gioca utilizzata per runnare test: ritorna l'efficacia dell'attacco (caselle colpite/tiri totali).

3 Testing

Per verificare la correttezza del gioco sono stati introdotti due file test.

- Test 5: verifica su un campione di n partite (letto da stdin) quante sono vinte dal computer, quante dall'umano e quante partite non vengono concluse a causa di errori: su un campione di 100000 partite non si sono verificati errori e chi inizia per primo ha più probabilità di vincere.
- Test 6: determina l'efficacia dell'attacco(colpite/totali) su un campione di n partite (letto da stdin).

Per i test 5 e 6 vengono utilizzate le funzioni gioca1 e gioca2 di cui sopra è descritto il comportamento. Inoltre, per non avere printf() o scelte da stdin durante la partita, si è provveduto a scrivere un file bnavale_muta.c, che prende condizioni di default fornite dai docenti (griglia 10x10, 2 navi da uno, 3 da due e 4 da tre, inserimento randomico, umano gioca per primo) ed evita la stampa di griglie ed esiti di tiro. I test vengono compilati con questo file, contenente anche le funzioni gioca1 e gioca2. Per provvedere ad avere il numero di colpite/mancate e tiri totali si sono aggiunti i campi hit, misses e shots alla struct area_t fornita dai docenti.

Infine, si è provveduto a creare un main gioca.c che gestisce un'intera partita. Il test4 non era sufficiente poiché non permetteva di passare alla funzione il numero di navi e le dimensioni della griglia da stdin.