

RELAZIONE PROGETTO S.O
A.A 2023-2024
VR472547
ALEX MERLIN

INTRODUZIONE

I file eseguibili verranno generati nella cartella bin, lanciando il comando make dalla cartella principale

SCELTE IMPLEMENTATIVE

- la coordinata da inserire per effettuare una mossa deve essere un solo numero da 1 a 9. Le caselle prendono valore crescenti da sinistra a destra;
- Allo scadere del timer il giocatore cede il turno all'avversario senza effettuare nessuna mossa;
- Nel caso di pareggio i giocatori vengono avvisati e la partita ricomincia in automatico;
- Nel caso di vittoria il vincitore può decidere se giocare o meno un'altra partita;
- Il giocatore che chiude il Terminale perde per abbandono
- Quando viene eseguito il client automatico serve inserire un timer maggiore di 5. Questo perchè per rendere la partita più fruibile si è deciso di rendere più lenta la generazione della mossa del computer attraverso la funzione sleep();

INIZIALIZZAZIONE

Una volta in esecuzione il server inizializza le seguenti variabili in memoria condivisa e semafori:

- Semid(key 100): semaforo composto da 3 elementi che mi vanno a rappresentare rispettivamente il server, il giocatore 1 e il giocatore 2;
- Mossa(key 110) : variabile in cui verrà salvata la mossa fatta da un giocatore;
- Count_move(key 130): variabile contatore per il numero di mosse;
- Matrix(key 99) : matrice di gioco;
- term1(key 103): stringa per comunicare con giocatore 1;
- term2(key 104): stringa per comunicare con giocatore 2;
- p1(key 101): struttura contenente informazioni giocatore 1;
- p2(key 102): struttura contenente informazioni giocatore 2;

Poi attende il primo giocatore.

Eseguendo il client 2 volte verranno collegati rispettivamente i due giocatori alle variabili condivise e ai semafori già creati in precedenza.

SEZIONI RILEVANTI DEL PROGRAMMA

CREAZIONE E LIBERAZIONE MEMORIA CONDIVISA

Ogni volta che viene creata una variabile condivisa, il suo *id* ed *indirizzo* vengono salvate rispettivamente in due array globali.

Così ogni qual volta il programma si chiuda, basta accedere sequenzialmente a questi array per scollegare e liberare la memoria.

INSERIMENTO MOSSA

Il Client prima di richiedere la mossa da inserire al giocatore, genera un processo figlio.

Mentre il padre attende, il processo figlio si occupa di impostare il timer per la mossa e attende la scelta dell'utente. Se scade il timer, il figlio esegue una exit con argomento 2.

L'argomento di uscita del figlio viene visionato dal padre attraverso la macro

WEXITSTATUS. In caso di riscontro(= 2) il giocatore cede il turno all'avversario. Altrimenti il client si occupa di salvare la *mossa* nella matrice, se valida, o impostare *mossa* con un valore sentinella, per capire per quali motivi la mossa non è valida:

- mossa = -2 : è stata scelta una posizione già occupata
- mossa = -3 : è stato inserito un valore diverso da 0-9.