

Realizzare questo progetto ci ha visto portare a compimento un lungo percorso di progettazione. Abbiamo deciso di lavorare parallelamente, grazie ad una repository di Git, seguendo due strade distinte. La prima relativa alla suddivisione dei task e all'intercomunicazione tra i processi; la seconda più progettuale ed algoritmica. A lavoro completato abbiamo poi collaborato attivamente per risolvere i problemi che si sono man mano presentati. È risultato cruciale un ambiente preimpostato condiviso e il massiccio debug sia manuale che assistito (gdb).

Introduzione

Oltre ai tre tipi di processo principali (Master, Player e Piece), il progetto è diviso in diversi moduli che raggruppano strutture e funzioni utili alla specifica componente del programma (come, ad esempio, Table che contiene il codice necessario per operare sulla tabella di gioco). Durante l'esecuzione il programma attraversa principalmente tre momenti:

- L'**avvio del programma**, in cui il processo Master si occupa di creare alcune strutture fondamentali, handler per SIGINT e SIGALARM, gli oggetti di sincronizzazione necessari e la memoria condivisa che costituirà la scacchiera. Successivamente vengono creati i Player che a loro volta generano le pedine (Piece). Ciascun processo applica poi le maschere ai segnali e rimane in attesa di comandi da parte del Master.
- L'**inizializzazione del round**, suddiviso in tre sottofasi coordinate dal Master ed implementate in ogni processo principale in maniera differente. La fase 1 prevede il posizionamento delle pedine sulla tabella. Nella fase successiva avvengono poi il posizionamento delle bandierine da parte del Master e l'assegnazione del bersaglio a ciascuna pedina da parte dei Player. L'ultima fase vede il Master avviare il timer e dare il via alle pedine.
- Il **round**, nel quale le pedine si muovono ciascuna in autonomia tentando di catturare le bandiere in campo. Il Master rimane vigile per tutta la durata in attesa della cattura delle bandiere. Il round può terminare per mancanza di bandierine in campo o per scadenza del timer.

Dettagli implementativi

Master

Il Master è il processo che si occupa della configurazione iniziale e di gran parte della sincronizzazione. In ogni round il Master genera e posiziona casualmente le bandierine, abilita il passaggio di fase in fase, rimuove le bandierine catturate assegnando il punteggio ai Player, attiva il restart ed infine gestisce la terminazione del gioco.

Player

I Player svolgono la preliminare funzione d'inizializzare correttamente le pedine, posizionandole in modo casuale sulla scacchiera prima che il Master abbia già piazzato le bandierine. Successivamente il processo si occupa di ricercare ed inviare a ciascun pezzo un target per il round corrente.

Piece

Una volta che una pedina ha ricevuto gli ordini dal proprio giocatore si mette in attesa dell'inizio round. Allo scattare dell'ordine del Master, le pedine iniziano a muoversi considerando come bersaglio la bandiera indicatagli nell'ordine ricevuto. Ciascuna pedina si muove conoscendo esclusivamente l'obiettivo e la cella precedentemente occupata, senza pertanto sapere la presenza di ostacoli lungo il percorso. Ad ogni modo una pedina può decidere di cambiare target qualora ritenga che sia irraggiungibile a causa di troppe celle bloccate lungo il percorso o per un insufficiente numero di mosse. Le strategie di movimento sono composte da combinazioni ripetute del singolo movimento sull'asse verticale o orizzontale nel modo seguente:

- X_BEFORE: privilegia prima movimento sull'asse orizzontale;
- Y_BEFORE: privilegia prima il movimento sull'asse verticale;

- **DIAGONAL**: alterna un movimento su asse verticale e uno orizzontale;
- **CHAOS_THEORY**: sceglie casualmente l'asse su cui muoversi ad ogni movimento;

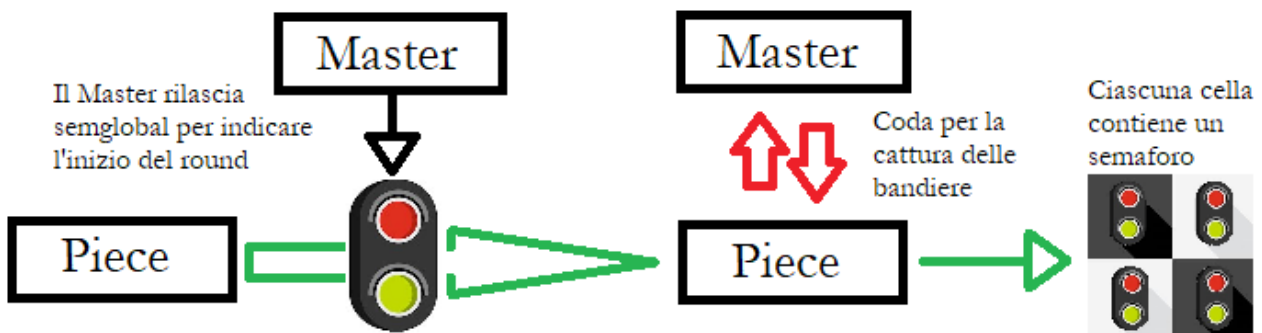
Nel caso in cui incontri un ostacolo lungo il percorso, la pedina avvia una speciale manovra evasiva che sovrascrive momentaneamente il target con una posizione libera, possibilmente nella stessa direzione di quest'ultimo e che non preveda di ripercorrere la stessa strada da cui la pedina è venuta.

Sincronizzazione

I processi si sincronizzano per lo più durante l'inizializzazione del round, a causa del prosieguo di fase in fase. Infatti, il cambio avviene solo quando il processo padre conferma il passaggio alla fase successiva. Si tratta pertanto di un cambio sequenzialmente gerarchico che fa capo al processo Master. La comunicazione con quest'ultimo avviene per lo più grazie al canale **MASTERCHANNEL** di una queue condivisa con Player e Piece. Sostanzialmente una fase inizia con l'invio di un messaggio di conferma a ciascun Player, che a loro volta girano lo stesso messaggio (attraverso una coda dedicata, chiamata **Key_MO**) alle pedine. Quando tutte le pedine di un Player hanno completato la fase lo rendono noto a quest'ultimo che completa a sua volta la fase con l'invio di un messaggio al Master. Solo quando tutti i Player hanno mandato questa conferma il Master procede con l'inizio del momento successivo. Al termine di una fase, i Player entrano in una funzione (detta **stand**) nella quale attendono per l'appunto il permesso dal Master di passare alla fase successiva, od eventuali comandi di restart qualora il round sia già in esecuzione. Esiste una funzione analoga per Piece (**getplay**).



Nell'ultima fase, il Master rilascia un semaforo (**semglobal**) sul quale le pedine si sono addormentate alla fine della fase 2. Questa release è il momento in cui inizia il round vero e proprio. Le pedine si muovono dunque sulla scacchiera in modo non sincronizzato e comunicano con il Master direttamente solo per indicare la cattura di una bandiera. Essendo la scacchiera un frammento di **shared-memory** condivisa a tutti ciascuna cella "contiene" un semaforo che indica l'occupazione della stessa. Per accedere ad una cella una pedina deve abbassare correttamente il semaforo corrispondente e solo allora potrà effettuare il movimento vero e proprio (alzando al termine il semaforo della cella appena abbandonata).



Al termine di un round per mancanza di bandierine, il Master interrompe il movimento delle pedine rialzando il semaforo **semglobal**. A questo punto, attraverso un fase supplementare (**RESTART**) un nuovo round viene inizializzato a partire dalla fase 2 (successiva alla creazione e al posizionamento delle pedine).

Il gioco termina quando viene ricevuto un **SIGALARM**, il quale attiva una chiusura a cascata di tutti i processi e relativi IPC.