

TRACCIA B – Conquista del territorio

Descrizione Sintetica

Si realizzi un sistema client–server in cui più giocatori cercano di conquistare il maggior numero possibile di celle di una mappa. Ogni cella appartiene all’ultimo giocatore che vi è passato. Gli ostacoli della mappa devono invece essere scoperti individualmente tramite esplorazione. Il progetto va sviluppato in C su piattaforma UNIX con comunicazione tramite socket TCP e correddato da documentazione.

Descrizione Dettagliata

La mappa è una griglia composta da celle libere e celle muro. Quando un giocatore entra in una cella libera, quella cella diventa sua: il server la marca come appartenente a quel giocatore. Se un altro giocatore passa successivamente sulla stessa cella, la conquista viene ribaltata. Ogni giocatore inizia posizionato in una cella libera casuale. Le celle muro non sono note in anticipo: ciascun giocatore scopre gli ostacoli solo nelle zone che esplora. Gli ostacoli rimangono quindi “privati” a ogni giocatore (fog-of-war sugli ostacoli), mentre le celle conquistate sono pubbliche. La partita termina dopo un timeout globale. Vince il giocatore che, alla fine, possiede il maggior numero di celle.

Per accedere al gioco ogni utente dovrà prima registrarsi al sito indicando password e nickname. Non c'è un limite a priori al numero di utenti che si possono collegare con il server.

Il server mantiene: la mappa completa degli ostacoli, la mappa globale della proprietà delle celle, lo stato di ogni giocatore (posizione, celle muro scoperte). Quando un giocatore esegue un movimento, il server aggiorna la sua posizione, rivela eventuali ostacoli attorno a lui e gli invia una **mappa locale** che contiene solo: gli ostacoli da lui realmente scoperti, e le informazioni di proprietà delle celle incluse nella finestra locale. Periodicamente (ogni T secondi) il server invia a *tutti* i client: la **mappa globale della proprietà** (chi possiede cosa), le **posizioni correnti dei giocatori**.

Il server deve gestire più client contemporanei, ricevere comandi, accettare nuove connessioni e inviare aggiornamenti periodici senza bloccare.

Il client consentirà all'utente di collegarsi ad un server di comunicazione, indicando tramite riga di comando il nome o l'indirizzo IP di tale server e la porta da utilizzare. Una volta collegato ad un server l'utente potrà accedere al servizio come utente con un nickname. Il client permetterà all'utente di: spostarsi sulla mappa, disconnettersi, vedere la lista degli utenti collegati, visualizzare la mappa locale e, periodicamente, la mappa globale inviata dal server.

Regole generali

Il server ed il client vanno realizzati in linguaggio C su piattaforma UNIX/Linux. Le comunicazioni tra client e server si svolgono tramite socket TCP.

Oltre alle system call UNIX, i programmi possono utilizzare solo la libreria standard del C. E' sconsigliato l'uso di primitive non coperte dal corso al posto di quelle studiate. Il server non deve inviare alcun output su standard output, non deve ricevere nessun input da standard input e può inviare output su standard error solo in caso di terminazione.

Relazione

Il progetto va accompagnato da una relazione che contenga almeno le seguenti sezioni:

1. Una guida d'uso per il server e per il client, che illustri le modalità di compilazione e d'uso dei due programmi.
2. Una sezione che illustri il protocollo al livello di applicazione utilizzato nelle comunicazioni tra client e server (non il protocollo TCP/IP!).

3. Una sezione che descriva i dettagli implementativi giudicati più interessanti, eventualmente corredati dai corrispondenti frammenti di codice.
4. La relazione dovrebbe constare di circa 10 pagine. Indicare sulla copertina della relazione i componenti del gruppo.

Consegna del progetto

Entro la data prescelta per lo scritto finale (con eccezione per il primo appello) devono essere consegnati al docente il progetto e la relazione. Il progetto e la relazione saranno inviati via chat su teams in un archivio compresso in formato zip o rar. Durante l'esame orale, il client ed il server verranno testati, eseguendoli su due o più macchine diverse.