

Descrizione Sintetica

Realizzare un sistema client-server che consenta a più utenti di giocare ad un gioco di guerra tra due eserciti. Si utilizzi il linguaggio C su piattaforma UNIX. I processi dovranno comunicare tramite socket TCP. Corredare l'implementazione di adeguata documentazione.

Descrizione Dettagliata

Il server manterrà una rappresentazione dell'ambiente in cui verranno posizionati i personaggi, gli ostacoli ed altri elementi del gioco. L'ambiente sia rappresentato da una matrice in cui gli utenti si potranno spostare di un passo alla volta nelle quattro direzioni: S, N, E, O. Il server posizionerà nella matrice ostacoli (rappresentati opportunamente, ad esempio con numeri: 0 libero, 1 ostacolo) in posizione random. Ogni esercito dovrà difendere una propria torre e abbattere quella del nemico: la torre dell'esercito 1 sarà posizionata in alto a sinistra, quella dell'esercito 2 in basso a destra. Ogni utente, una volta connesso al server, potrà partecipare al gioco con un soldato, indicando l'esercito di appartenenza, il tipo di soldato, e una posizione (x,y) sulla mappa nella zona del proprio esercito (1 nella metà a sinistra, 2 in quella a destra). L'utente potrà scegliere un soldato attaccante o un difensore. Al momento della creazione l'attaccante ed il difensore riceveranno rispettivamente 1000 e 3000 punti. Per fare un passo l'utente spenderà 1 punto e riceverà l'informazione sull'effetto proprio movimento: se lo spostamento porta ad una collisione con un ostacolo o una torre amica il movimento avrà effetto nullo; se lo spostamento porta nella locazione di un altro utente nemico o una torre nemica, l'utente nemico o la torre perderanno punti, 500 punti se il personaggio è attaccante, 200 punti se è difensore. Un soldato senza più punti sarà eliminato dal gioco. Quando un utente avrà abbattuto una torre nemica, o alla scadenza di un limite di tempo fissato, il server notificherà a tutti gli utenti del gioco il risultato finale, la fine della sessione e ne genererà una nuova. Nel caso di fine per scadenza del tempo verrà notificato a tutti un pareggio.

Per accedere al servizio 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 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: registrarsi come nuovo utente o accedere al servizio come utente registrato. Il servizio permetterà all'utente di: spostarsi di una posizione, disconnettersi, vedere la lista degli utenti collegati e la squadra di appartenenza, vedere il tempo mancante, vedere la posizione di tutti i soldati della propria squadra indicando il tipo ed i punti rimanenti; vedere la posizione degli ostacoli e incontrate da tutti i personaggi dall'inizio del gioco.

Il server dovrà supportare tutte le funzionalità descritte nella sezione relativa al client. All'avvio del server, sarà possibile specificare tramite riga di comando la porta TCP sulla quale mettersi in ascolto. Il server sarà di tipo concorrente, ovvero è in grado di servire più client simultaneamente. Durante il suo regolare funzionamento, il server effettuerà il logging delle attività principali in un file apposito. Ad esempio, memorizzando la data e l'ora di connessione dei client e il loro nome simbolico (se disponibile, altrimenti l'indirizzo IP), e la data e l'ora delle eliminazioni o della conquista delle torri.

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 (ad es., code di messaggi) 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 che tipo di dati si scambiano il client e server durante le diverse fasi del gioco.
3. Una sezione che descriva i dettagli implementativi giudicati più interessanti (con particolare riferimento alle system call oggetto del corso), eventualmente corredati dai corrispondenti frammenti di codice.
4. In appendice, la relazione deve riportare il codice sorgente integrale del progetto. Orientativamente, la relazione dovrebbe constare di almeno 10 pagine, esclusa l'appendice. 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), vanno consegnati al docente il progetto e la relazione. Il progetto e la relazione vanno inviati all'indirizzo alberto.finzi@unina.it in un archivio compresso in formato zip o rar. La relazione va consegnata al docente anche in formato cartaceo al momento della discussione. Durante l'esame orale, il client ed il server verranno testati, eseguendoli su due o più macchine diverse.