

Progetto “Blackjack”

Descrizione sintetica

Realizzare un sistema client-server che consenta a più giocatori di giocare a Blackjack contemporaneamente, contro lo stesso banco. Si utilizzi il linguaggio C su piattaforma UNIX. I processi comunicano tramite socket TCP. Corredare l'implementazione di adeguata documentazione.

Descrizione dettagliata

Il Blackjack è un gioco d'azzardo basato sulle carte francesi. Le carte da 2 a 10 valgono quanto il loro numero. Le figure valgono 10. L'asso vale 11, tranne quando appartiene ad una mano di valore maggiore di 21, nel qual caso vale 1. I giocatori partono con un patrimonio iniziale di 10000 fiches. All'inizio di ogni partita, ciascun giocatore dichiara una puntata iniziale di 5 fiches. Il banco distribuisce due carte scoperte a ciascun giocatore e due carte a se stesso, di cui una coperta ed una scoperta. A quel punto, i giocatori effettuano a turno le loro mosse. Le mosse possibili sono:

- Chiedere una carta, che sarà fornita dal banco scoperta. Se il totale delle carte ricevute è superiore a 21, il giocatore ha perso la partita corrente e il turno passa al giocatore successivo, altrimenti il turno resta al giocatore corrente.
- Aumentare la posta in gioco. Il giocatore indica di quante fiches vuole aumentare la posta. Il turno resta al giocatore corrente.
- Fermarsi. Questa mossa conclude la giocata e fa passare il turno al giocatore successivo.

In qualunque momento, un giocatore può collegarsi alla partita che è attualmente in corso, diventando l'ultimo giocatore della mano.

Quando l'ultimo giocatore della mano ha finito di muovere, tocca al banco. Il banco scopre la sua carta coperta e poi pesca nuove carte finché il suo totale è almeno pari a 17. Se il banco “sballa” (cioè supera 21), tutti i giocatori che non hanno sballato ricevono una vincita pari a quanto hanno puntato in quella mano. Altrimenti, ricevono la vincita tutti quelli che hanno un punteggio strettamente superiore al banco. Quelli che hanno un punteggio pari a quello del banco non vincono né perdono nulla, mentre quelli che hanno un punteggio inferiore perdono la somma che hanno puntato.

Il client consente all'utente di collegarsi ad un server di gioco, 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 dovrà fornire un nickname. Poi, l'utente viene inserito nella partita corrente, come ultimo giocatore.

Quando non è il suo turno, il giocatore vede tutte le mosse degli altri giocatori e del banco, e non può interagire col gioco. Arrivato il suo turno, il giocatore potrà scegliere le mosse tra quelle illustrate sopra. Se un giocatore esaurisce le fiches, viene scollegato dal gioco.

Il server deve supportare tutte le funzionalità descritte nella sezione relativa al client.

All'avvio del server, è possibile specificare tramite riga di comando la porta TCP sulla quale mettersi in ascolto, nonché la durata massima di una partita. Il server è di tipo concorrente, ovvero è in grado di servire più client simultaneamente.

Nel dare le carte, il server non deve semplicemente generare carte a caso, ma deve simulare correttamente il comportamento di *quattro* mazzi di carte francesi mischiati a caso. Quando, alla fine di una partita, le carte rimaste nel mazzo sono meno di 52, il server ricomincia con quattro mazzi mischiati e notifica i giocatori di questo evento. Avendo a disposizione un numero limitato di carte, il server non deve permettere a più di 20 giocatori di essere collegati contemporaneamente.

Durante il suo regolare funzionamento, il server effettua il logging delle attività principali in un file apposito. Ad esempio, memorizza 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 di creazione e di terminazione delle partite.

Opzionale: per evitare che un client indisciplinato blocchi tutta la partita, il server deve operare in modo che se un client impiega più di 2 minuti per scegliere la sua mossa, è come se avesse scelto di fermarsi.

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. Sarà valutato negativamente 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 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 (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 il numero assegnato al gruppo e i componenti del gruppo.

Consegna del progetto

Entro la data prescelta per lo scritto finale, vanno consegnati al docente il progetto e la relazione. Il progetto e la relazione vanno inviati all'indirizzo faella.didattica@gmail.com in un archivio compresso in formato zip o rar. La relazione va consegnata al docente *anche* in formato cartaceo. Durante l'esame orale, il client ed il server verranno testati, eseguendoli su due o più macchine diverse.