

Progetto di Sistemi Operativi - A.A. 2009/2010

Ivan Lanese

April 21, 2010

1 Funzionalità del programma

Si deve implementare un'architettura client-server per il gioco online del tris (vedere [http://it.wikipedia.org/wiki/Tris_\(gioco\)](http://it.wikipedia.org/wiki/Tris_(gioco)) per informazioni sul gioco). Il progetto comprende la realizzazione sia del server che del client.

Il client fornisce l'interfaccia per consentire all'utente di interagire col server. All'avvio il client deve leggere un file di configurazione dove vengono fissati i suoi parametri di funzionamento. L'interazione successiva avviene attraverso comandi immessi dall'utente.

Il server gestisce la creazione di nuove partite, le partite in corso e la classifica dei giocatori. Il server è concorrente, quindi può gestire più client e più partite allo stesso momento. All'avvio il server deve leggere un file di configurazione dove vengono fissati i parametri di funzionamento. L'unica possibile interazione col server è il comando "quit", che comporta l'interruzione del servizio.

2 Client

Appena avviato il client legge un file di configurazione che determina alcuni parametri di funzionamento:

- indirizzo e porta del server a cui connettersi;
- nome utente e password per l'accesso al server.

Dopo l'inizializzazione il client si connette al server usando il nome utente e la password letti dal file di configurazione. A questo punto l'utente può usare il client per eseguire le seguenti operazioni:

- richiedere la lista degli utenti connessi al server;
- creare una nuova partita, con sé stesso come giocatore 1, e mettersi in attesa del giocatore 2;

- richiedere la lista delle partite in attesa del giocatore 2;
- far partire una delle partite in attesa, collegandosi come giocatore 2;
- richiedere la lista delle partite in corso;
- diventare spettatore di una partita in corso;
- richiedere la classifica;
- richiedere l'help con le istruzioni per l'uso del client;
- terminare l'esecuzione del client.

Il client esegue tutte le operazioni (tranne la stampa dell'help) interagendo col server, poiché tutte le informazioni di stato (utenti connessi, informazioni sulle partite in attesa e su quelle in corso, ...) risiedono sul server. Il client deve anche comunicare al server la sua intenzione di terminare.

Le interazioni tra utente e client avvengono tramite input/output da tastiera (la scelta delle modalità esatte è lasciata agli studenti). L'interfaccia del client può essere semplicemente di testo, non è necessario implementare un'interfaccia grafica. L'implementazione dell'interfaccia grafica di per sé non dà diritto ad una valutazione più alta.

Nell'implementazione del client vanno gestite le possibili situazioni di errore, quali terminazione del server, comandi errati nel file di configurazione o durante l'interazione con l'utente e altro. Eventuali errori non devono, ove possibile, compromettere il funzionamento globale del programma, e devono comunque essere diagnosticati all'utente da opportuni messaggi d'errore chiaramente interpretabili e specifici (**Errore** è troppo generale...). Non devono essere considerate nel progetto problematiche legate alla sicurezza (ad es., crittografia per le password).

3 Partite

Una partita inizia quando due giocatori (il creatore ed un altro) sono collegati ad essa. La partita procede con mosse alternate dei 2 giocatori fino alla vittoria di uno dei 2, al pareggio, o finché uno dei 2 non si arrende. La resa è fatta tramite apposito comando dal client. Chiaramente entrambi i giocatori devono vedere lo stato della partita, ma tutti i controlli (mossa legale, vittoria, ...) vengono fatti dal server. Al termine della partita il server comunica il risultato finale (vittoria/sconfitta/pareggio) ai client e aggiorna le statistiche. Le statistiche registrano per ogni giocatore il numero di partite effettuato, il numero di vittorie e il numero di pareggi.

Un utente può entrare in una partita in corso come spettatore. In questo caso vede le mosse fatte dai 2 giocatori (in tempo reale, nei limiti del possibile). Gli spettatori non possono interagire con la partita.

4 Server

Il server gestisce l'elenco degli utenti connessi, l'elenco delle partite in attesa del secondo giocatore, l'elenco delle partite in corso, le singole partite in corso e la classifica.

Il server è configurabile tramite un file che specifica (almeno) la porta di ascolto del server. L'unico comando che il server accetta è "quit", che ne causa la terminazione. Non è richiesto nessun altro meccanismo di interazione col server per regolarne il funzionamento. Il server riceve richieste di connessione dai client. I client sono identificati da login e password (si assume che i login siano tutti diversi tra loro). Dopo il login il server gestisce le richieste del client, come descritto nella parte relativa al client.

Il server gestisce inoltre le statistiche sulle partite. In particolare, per ogni utente, ricorda il numero di partite giocate, il numero di vittorie e il numero di pareggi. Il punteggio di un giocatore è dato da $1000 * (\text{vittorie} + (\text{pareggi}/2)) / \text{giocate}$. La classifica è l'elenco degli utenti ordinato per punteggio.

In caso di terminazione (richiesta tramite il comando "quit") le partite in corso vengono interrotte. Devono essere preservate le statistiche sulle partite (escluse quelle interrotte).

5 Gruppi

I gruppi dovranno essere composti da minimo 3 - massimo 5 persone. I gruppi che intendono svolgere questo progetto devono necessariamente COMUNICARE PER EMAIL (lanese@cs.unibo.it) - con oggetto "gruppo sistemi operativi" - LA COMPOSIZIONE DEL GRUPPO entro e non oltre LUNEDÌ 1 MAGGIO 2010. Nella composizione del gruppo, si devono specificare per ogni componente NOME-COGNOME-NUMERO DI MATRICOLA. Inoltre si deve specificare un NOME DEL GRUPPO di 6 digit (lettere o numeri, ad esempio le iniziali di nomi e cognomi) e un EMAIL DI RIFERIMENTO alla quale invierò eventuali comunicazioni (quale la data della discussione).

Coloro che non riescono a costituire un gruppo come richiesto devono comunicarmi via mail la loro situazione (i loro dati, nome-cognome-matricola-email, ed eventuali preferenze legate al luogo di lavoro e ai tempi di lavoro, per aiutarmi a creare gruppi uniformi). Provvederò io a raggrupparli. I gruppi così creati terranno conto delle preferenze espresse solo ove possibile, e non sarà possibile modificarli. Anche qualora i gruppi così creati includano persone residenti in posti diversi, tali persone si devono organizzare per gestire il lavoro di gruppo.

I singoli studenti devono inoltre iscriversi all'appello di esame all'indirizzo:
<http://esami.int.nws.cs.unibo.it/cgi-bin/Main.php>

Da notare che l'appello è unico, indipendentemente dalla data di consegna del progetto e dalla data effettiva della discussione.

Coloro che non daranno comunicazione della loro condizione di gruppo entro il 1 MAGGIO NON POTRANNO FARE IL PROGETTO.

6 Modalità di consegna

Si devono consegnare due file: un archivio .tar.gz o .zip che contenga tutti i file relativi al progetto e un file README che specifichi passo per passo come, a partire dall'archivio, si possano eseguire tutte le operazioni necessarie per compilare e eseguire sia il server che il client. L'archivio deve contenere le seguenti directory:

src, contenente i sorgenti del progetto. Si ricorda che i sorgenti devono essere scritti seguendo opportune regole di coding style e devono essere commentati in maniera chiara e non banale. Es. di commento banale:

```
i++; // incremento della variabile
```

Es. di commento appropriato:

```
// ricerca della prima posizione libera
while ((tree[firstFreePosition] != null) && (i < size))
{
    firstFreePosition = (firstFreePosition + 1) % size;
    i++;
}
if (tree[firstFreePosition] != null)
{ firstFreePosition = -1; }
}
```

classes, contenente i file .class con il bytecode risultante dalla compilazione dei sorgenti

doc, contenente la relazione (circa 8-10 pagine) sul progetto (in formato DOC o PDF) nel file project.doc o project.pdf, dove vengono descritte le scelte progettuali effettuate (per esempio: uso di una certa struttura dati piuttosto che un'altra, descrizione per sommi capi della struttura del progetto, del codice sorgente, dei metodi più importanti e considerazioni sui problemi principali riscontrati). Specificare nella relazione i componenti del gruppo e le relative email.

Per la consegna dei file si deve inviare una email con subject "consegna progetto <nomeGruppo>" e in attachment i file richiesti a lanese@cs.unibo.it. Nella email vanno specificati chiaramente i componenti del gruppo con le relative email; va specificata inoltre l'email di riferimento per comunicazioni successive al gruppo.

La documentazione deve essere consegnata anche in formato cartaceo lasciandola nella casella di posta del prof. Sangiorgi (che si trova al piano terra del dipartimento di informatica, poco prima dell'ufficio). Vanno consegnate in forma cartacea i seguenti files:

- project.doc (o .pdf);

- tutti i file sorgenti stampati in orizzontale (sotto linux si può usare `enscript -r nomefile`). Ogni file deve essere graffettato nell'angolo in alto a sinistra.

In seguito verrà fissata la data e l'ora in cui presentarsi per la discussione del progetto.

NOTA: la relazione del progetto è molto importante e progetti con relazioni scarse (ad esempio di una facciata) possono essere anche solo per questo motivo giudicati insufficienti. La relazione non deve contenere un semplice elenco di classi o peggio la ripetizione delle specifiche, ma le idee e le scelte progettuali, soprattutto per quelli considerati punti chiave. In questo senso è gradita anche la sintesi: relazioni di 60 pagine sono ugualmente discutibili di quelle di 2 pagine. Una buona idea può essere di scrivere la relazione durante lo sviluppo del progetto, segnando di volta in volta le cose importanti di cui si discute, e mettendola a posto alla fine.

7 Date di consegna

Ci sono due date di consegna:

-all'inizio di Luglio, alle 24 di Venerdì 2 Luglio 2010: chi consegna il progetto entro questa data farà presumibilmente la discussione entro il mese di Luglio (dipendentemente dai tempi di correzione).

-alla fine di Settembre, alle 12 di Venerdì 24 Settembre 2010

In seguito a tali date verrà fissata la data e l'ora in cui presentarsi per la discussione del progetto e il referente del gruppo riceverà una comunicazione via email.

8 Valutazione del progetto

Il progetto verrà valutato mediante una discussione di gruppo, con tutti i componenti del gruppo presenti. La discussione, e il testing del progetto, sarà fatta nei laboratori ERCOLANI. Si richiede quindi che il progetto sviluppato funzioni sulle macchine dei laboratori Ercolani sulle quali verrà testato. INVITO QUINDI TUTTI A TESTARE IL PROPRIO SOFTWARE SU TALI MACCHINE e ad imparare a lavorare e a muoversi anche su di esse.

Non è possibile per i componenti di un gruppo effettuare la discussione in incontri separati. Al termine della discussione ad ogni singolo componente verrà assegnato un voto in base all'effettivo contributo dimostrato nel lavoro di progetto. La valutazione del progetto è indipendente dal numero di persone che compongono il gruppo.

9 Note importanti

1. Il progetto deve funzionare: non devono essere presentati progetti non funzionanti, perchè non verranno presi in considerazione.

2. Il progetto deve fare "qualcosa": un progetto funzionante che scrive solo "hello world" non può essere accettato.
3. Non si accettano (e fanno pessima impressione) email o richieste di eccezioni sui progetti con motivazioni legate a esigenze di laurearsi o di non voler pagare le tasse per un'altro anno: chi ha tali esigenze deve pensarci prima.
4. Chi copia o fa copiare, anche un sola parte del progetto, si vedrà invalidare completamente il progetto senza possibilità di appello. Dovrà quindi rifare un nuovo progetto l'anno successivo.

10 Domande sul progetto

Per le domande sul progetto si consiglia di usare il newsgroup del corso:

`unibo.cs.scienzeinternet.so`

Siete invitati a rispondere alle domande dei vostri colleghi sul newsgroup. Risposte corrette verranno positivamente valutate in sede di esame.

Solo in situazioni eccezionali è possibile chiedere informazioni mandando email direttamente a me (lanese@cs.unibo.it) o venire a ricevimento (previo appuntamento preso via mail, che specifichi la motivazione della richiesta).