

DESCRIZIONE GENERALE:

L'elaborato è diviso in due file sorgente: `GameServer.c` e `gamer.c`. Il primo è il file del banco e gestisce il gioco e i giocatori. Il secondo è il file sorgente di ogni giocatore.

Il gioco inizia lanciando il comando

```
./GameServer numeroGiocatori sommaBanco : dove il primo parametro
specifica il numero di giocatori (che deve essere maggiore di 0) e il
secondo è il numero iniziale di monete del banco (che deve essere un
numero intero maggiore di 0).
```

Questo comando fa partire l'esecuzione del banco che, inizialmente, si mette in attesa. A questo punto i giocatori possono unirsi al gioco.

Ogni giocatore si unisce al gioco lanciando il comando

```
./gamer nomeGiocatore somma : dove il primo parametro è il nome del
giocatore e il secondo è il numero iniziale di monete (che deve essere un
numero intero maggiore di 1).
```

Questo comando aggiunge il giocatore al gioco e lo mette in pausa.

Quando il numero prestabilito di giocatori è stato raggiunto, il banco si risveglia e fa iniziare la partita.

A turno il banco risveglia ogni giocatore in attesa e si rimette in pausa. Risvegliandosi, ogni giocatore punta la propria scommessa che è un numero intero casuale tra l'1% e il 50% del numero di monete attualmente possedute dal dato giocatore. Il valore ottenuto viene memorizzato nell'area di memoria 3 (v.d. memorie condivise e semafori, pag. 3). Successivamente, ogni giocatore risveglia il banco e si riaddormenta fino a che tutti i giocatori non hanno puntato una cifra.

Quando questo accade, il banco, risvegliato dall'ultimo giocatore, lancia i dadi (viene calcolato un numero intero casuale compreso tra 2 e 12) e ricomincia il giro dei giocatori svegliandoli uno ad uno e si riaddormenta. Ogni giocatore si risveglia e lancia a sua volta i dadi, risveglia il server e si addormenta. Il valore ottenuto sia dal server che dai giocatori lanciando i dadi viene memorizzato nell'area di memoria condivisa 2 (v.d. memorie condivise e semafori, pag. 3).

Quando è terminato nuovamente il giro, l'ultimo giocatore risveglia il server che determina il vincitore, cioè quale giocatore ha ottenuto il numero più alto lanciando i dadi. Il vincitore può essere:

- il banco: in questo caso tutte le monete scommesse dai giocatori si sommano alle monete del banco e i giocatori perdono le monete scommesse;
- il banco e uno o più giocatori: in questo caso il/i giocatore/i che ha/hanno ottenuto lo stesso punteggio dal lancio dei dadi riceve/ricevono dal banco la cifra che aveva/avevano scommesso raddoppiata, il banco riceve le monete scommesse dai

giocatori che hanno perso e i giocatori che hanno perso perdono le monete scommesse;

- uno o più giocatori: il/i giocatore/i che ha/hanno vinto riceve/ricevono dal banco la cifra che aveva/avevano scommesso raddoppiata, il banco riceve le monete scommesse dai giocatori che hanno perso e i giocatori che hanno perso perdono le monete scommesse.

Successivamente, il banco risveglia ad uno ad uno i giocatori settando la terza cella dell'area di memoria 2 (v.d. memorie condivise e semafori, pag. 3) ad un valore: se il giocatore ha vinto, il valore sarà -1 altrimenti sarà -2. Se, invece, ha vinto il banco il valore sarà 0.

Quando ogni giocatore si risveglia, controlla la cella e stampa a video un messaggio. Successivamente, risveglia il server e si riaddormenta.

Quando tutti i giocatori sono stati svegliati il server fa iniziare una nuova partita: fa scommettere i giocatori, lancia e fa lanciare i dadi ai giocatori e determina il vincitore.

Nota: l'addormentarsi/svegliarsi in questa parte del gioco è gestita decrementando/incrementando la cella corrispondente del semaforo 2 (v.d. memorie condivise e semafori, pag. 3).

Il gioco termina solo quando il banco va in bancarotta o viene terminato premendo CTRL+C. Se, invece, uno o più giocatori vanno in bancarotta o lasciano il gioco, questo si sospende fino a che non viene ristabilito il numero di giocatori necessario per giocare (si devono quindi collegare nuovi giocatori per sostituire i giocatori terminati).

I giocatori possono lasciare il gioco premendo CTRL+C solo tra una partita e l'altra. Quindi devono aspettare che la partita termini per poter abbandonare il gioco. Se al termine di una partita non hanno abbastanza soldi per continuare, i giocatori vengono interrotti automaticamente.

Quando il server e/o i giocatori terminano viene stampato un riassunto con le partite perse, vinte, giocate, il numero di monete iniziali e il numero di monete finali.

Quando il numero massimo di giocatori è stato raggiunto nessun altro giocatore può unirsi al gioco lanciando il comando `./gamer`. Il nuovo giocatore deve attendere che un giocatore termini o vada in bancarotta per prendere il suo posto e poter giocare.

MEMORIE CONDIVISE E SEMAFORI:

Il file sorgente del banco oltre che gestire il gioco crea le aree di memoria condivisa e i semafori utilizzati poi per l'esecuzione corretta delle diverse partite.

- Memoria condivisa:

Vengono allocate tre aree di memoria condivisa:

1. Memoria 1: è un vettore che contiene $n+2$ celle, dove n indica il numero di giocatori specificato nella riga di comando per lanciare il banco.

ID semaforo 1	Stato server	Stato giocatore 1	Stato giocatore 2	...	Stato giocatore n
---------------------	-----------------	-------------------------	-------------------------	-----	-------------------------

La prima cella contiene l'identificativo del semaforo 1, la seconda lo stato del server e le altre gli stati dei giocatori. Lo stato è un numero intero: 0 indica che il server o il giocatore sono terminati, 1 indica che il server o il giocatore sono attivi.

2. Memoria 2: è un vettore di $n+3$ celle.

ID semaforo 2	n	Lancio dadi banco	Lancio dadi giocatore 1	...	Lancio dadi giocatore n
---------------------	---	-------------------------	----------------------------------	-----	----------------------------------

La prima cella contiene l'identificativo del semaforo 2, la seconda contiene il numero di giocatori specificato da riga di comando, la terza contiene il numero ottenuto dal banco lanciando i dadi e le successive celle contengono il numero ottenuto dai diversi giocatori lanciando i dadi.

3. Memoria 3: è un vettore di n celle.

Scommessa giocatore 1	Scommessa giocatore 2	...	Scommessa giocatore n
--------------------------	--------------------------	-----	--------------------------

Ogni cella contiene il numero di monete scommesse da ogni giocatore.

- Semafori:

Sono creati due semafori:

1. Semaforo 1: è un vettore di $n+1$ celle.

Banco	Giocatore 1	...	Giocatore n
-------	-------------	-----	-------------

Sveglia/addormenta il banco o i giocatori quando uno di questi terminano.

2. Semaforo 2: è un vettore di $n+2$ celle.

Numero giocatori online	Banco	Giocatore 1	Giocatore 2	...	Giocatore n
-------------------------------	-------	----------------	----------------	-----	----------------

La prima cella contiene il numero di giocatori attualmente online così calcolato: quando un giocatore si connette al gioco, lanciando il comando `./gamer`, il valore della cella (inizializzato a 0) è incrementato di uno, quando un giocatore va in bancarotta o termina decrementa di uno il valore della cella del semaforo. Le altre celle svegliano/addormentano o il server o i giocatori durante il normale svolgimento della partita.

TERMINAZIONI:

La terminazione del banco e/o dei giocatori è gestita dai processi figli dei programmi principali: sia GameServer.c che gamer.c eseguono la System Call `fork()` per generare un figlio ciascuno.

Il figlio del banco gestisce le terminazioni dei giocatori. Il figlio del giocatore gestisce la terminazione del banco.

- Terminazione del banco:
quando un giocatore lancia il comando `./gamer` accede al gioco mediante il processo padre e genera un processo figlio (che condivide le aree di memoria e i semafori del padre) che, in parallelo, gestisce la terminazione del server. Appena il figlio viene creato, viene messo in pausa decrementando di uno la cella corrispondente del giocatore nel semaforo 1.

Quando il banco va in bancarotta, il processo padre del banco invia a sé stesso il segnale 12 mediante la System Call `kill()`. Questo segnale è intercettato da una funzione che viene eseguita anche quando viene premuto CTRL+C.

La funzione esegue i seguenti passaggi:

1. controlla se il banco è andato in bancarotta o è stato terminato. Nel primo caso la cella 1 dell'area di memoria 1 è settata a 1.
2. sveglia il figlio di ogni giocatore incrementando di uno il semaforo corrispondente.
3. termina il proprio figlio.
4. elimina i semafori e le aree di memoria condivisa.
5. termina.

Quando il figlio del giocatore è risvegliato dal banco, invia il segnale 10 al proprio padre che intercetta il segnale ed esegue una funzione che stampa a video il motivo per cui il banco è terminato (CTRL+C o bancarotta (stato del banco impostato a 1)), termina il proprio figlio e termina.

- Terminazione di uno o più giocatori:
quando viene lanciato il comando `./GameServer` viene creato un processo figlio (che condivide le aree di memoria e i semafori con il padre) per gestire la terminazione dei giocatori. Appena viene creato, il processo figlio è messo in pausa decrementando di uno la cella 0 del semaforo 1. Il processo padre del banco, invece, gestisce il gioco.

Quando una partita è terminata, un giocatore può andare in bancarotta (viene lanciato dal processo padre del giocatore a sé stesso il segnale 15) o premere CTRL+C per terminare il gioco. Quando uno dei due eventi accade, il

corrispondente segnale è intercettato dal processo padre del giocatore che esegue una funzione che:

1. stampa a video la causa della terminazione (bancarotta o CTRL+C).
2. modifica da 0 a 1 lo stato del giocatore nella cella dell'area di memoria 1 corretta.
3. decrementa di uno il numero di giocatori contenuto nella cella 0 dell'area di memoria 3.
4. sveglia il figlio del server incrementando di uno la cella 0 del semaforo 1.
5. termina il proprio figlio.
6. termina.

Quando il processo figlio del banco è risvegliato dalla terminazione di un giocatore, invia al processo padre il segnale 10. Il processo padre del banco intercetta il segnale ed esegue una funzione che stampa a video un messaggio per notificare che un giocatore ha abbandonato il gioco.

Successivamente, una nuova partita ricomincia e il banco controlla se tutti i giocatori sono online. Se il numero contenuto nella cella 0 dell'area di memoria 2 è diverso da n, allora il processo padre del banco viene messo in pausa decrementando di uno la cella 1 del semaforo 2. Tale processo sarà risvegliato da ogni nuovo giocatore in modo che il gioco possa riprendere con una nuova partita con n giocatori.