

## PCAD 24/25 - Laboratorio 4

L'obiettivo di questo laboratorio è di programmare un'applicazione distribuita di produttore/consumatore in una rete.

### 1 Esercizio

L'idea è di avere un server TCP che aspetta delle connessioni su un port scelto da voi e su questo server si possono connettere due tipi di clienti:

1. I clienti produttore che mandano delle stringe,
2. i clienti consumatore che ricevono delle stringe prodotte.

Il protocollo di comunicazione deve funzionare in questo modo:

- per i clienti produttore: dopo la connessione, il cliente manda un messaggio "producer\n" (le aspie non fanno parte dei messaggi) poi aspetta un messaggio dal server "okprod\n" e poi il cliente manda una stringa senza carattere '\n' in mezzo che finisce con '\n' e si scollega. La stringa prodotta viene aggiunta ad una struttura FIFO dal server.
- per i clienti consumatore: dopo la connessione, il cliente manda un messaggio "consumer\n" poi aspetta un primo messaggio dal server "okcons\n" e un altro messaggio senza carattere '\n' in mezzo che finisce con '\n' e che corrisponde alla stringa più vecchia contenuta nella struttura FIFO del server. La stringa consumata viene cancellata dalla struttura FIFO. Il client rimane connesso finché ottiene una stringa (se la struttura FIFO è vuota, aspetterà). Dopo avere ottenuto la stringa il cliente si scollega.

Dovete fare due versioni del server, una in cui la lunghezza della struttura FIFO non è limitata, ed un'altra dove è limitata. Per questo ultimo caso, un produttore non può aggiungere una stringa quando la struttura è piena e deve aspettare (senza scollarsi di potere inserire la stringa nella struttura).

Il vostro server deve potere accettare più connessioni di clienti allo stesso momento, deve proteggere l'accesso concorrente alla struttura FIFO e deve fare in modo che quando un cliente aspetta, il thread associato non è in attesa attiva.

Dovete anche programmare dei clienti per testare il comportamento del vostro server e può essere una buona idea, provare ad usare i clienti/server di altri gruppi di studenti.

### 2 Consegna

Per la consegna, creare uno zip con tutti i vostri file. Lo zip dovrà anche contenere un file partecipanti.txt dove gli nomi di chi ha partecipato alla consegna (questo anche se siete da solo a farla).