

## Verifica pratica di TPSIT: Node.js

Realizzare un sistema di messaggistica tra client mediante un server TCP. In particolare, ogni client può inviare messaggi a ogni altro client ad un determinato costo in € prestabilito. Una volta esaurito il saldo nel suo conto, il client non è più in grado di inviare messaggi. Tuttavia, il conto di un client può essere ricaricato. Per implementare questo sistema, si richiede che vengano aggiunte le seguenti proprietà al connection socket:

- id univoco
- saldo (il valore iniziale è 5 €)
- stato della connessione (true se connesso, false se disconnesso)
- timestamp dell'inizio della connessione

Il socket verrà poi inserito nella lista globale dei client, necessaria per implementare le funzionalità richieste.

Il protocollo di comunicazione prevede i seguenti comandi:

- **info**: restituisce l'id univoco assegnato al client, il suo indirizzo ip e numero di porta e la durata della connessione in secondi (**0,5 punti**)
- **credit**: restituisce il saldo attuale del client in € (**0,5 punti**)
- **list**: restituisce la lista completa dei client attualmente connessi, mostrando per ognuno l'id univoco, l'indirizzo ip, la porta e la durata della connessione in secondi (**1 punto**)
- **<clientid> <message>**: invia un messaggio a uno specifico client, purché sia ancora connesso; il costo del messaggio è di 1 centesimo di € per ogni carattere scritto; il server risponde con il costo del messaggio; se il credito del client è insufficiente, il messaggio non viene inviato, con feedback al mittente (**1,5 punti**)
- **bcast <message>**: il messaggio viene inviato a tutti i client, tranne che al mittente del messaggio; il costo del messaggio è di 1 centesimo di € per carattere, moltiplicato per il numero di destinatari del messaggio; il server risponde con il costo del messaggio; se il credito del client è insufficiente, il messaggio non viene inviato, avvisando con un feedback al mittente (**1,5 punti**)
- **recharge <clientid> <euro>**: ricarica il client identificato da **<clientid>** della somma specificata come parametro, che però deve essere prelevata dal credito di chi effettua la ricarica; l'operazione va a buon fine solo se il credito del client è sufficiente e se il client destinatario è attualmente connesso; in tal caso, il destinatario della ricarica riceverà un messaggio di notifica con l'importo della ricarica e il mittente che la manda; il mittente viene comunque avvisato dell'esito dell'operazione (**2 punti**)
- **account**: restituisce la lista degli addebiti e degli accrediti del client; per ogni elemento della lista è specificato l'importo (negativo se si tratta di un addebito) e l'id dell'altro client coinvolto nell'operazione; se l'operazione è l'invio di un messaggio di broadcast, indicare "bcast" al posto dell'id del client (**2 punti**)

**Importante:** il codice sorgente deve essere adeguatamente **commentato**, **comprensibile** e **ben organizzato**; in caso contrario può essere applicata una penalità fino a **-1 punto**. Consegnare il file sorgente nella **propria cartella del server** nominandolo come **CognomeNome-PraticoNode.js**, direttamente nella cartella principale (non in una sottocartella!).

Voto min = 1, voto max = 10