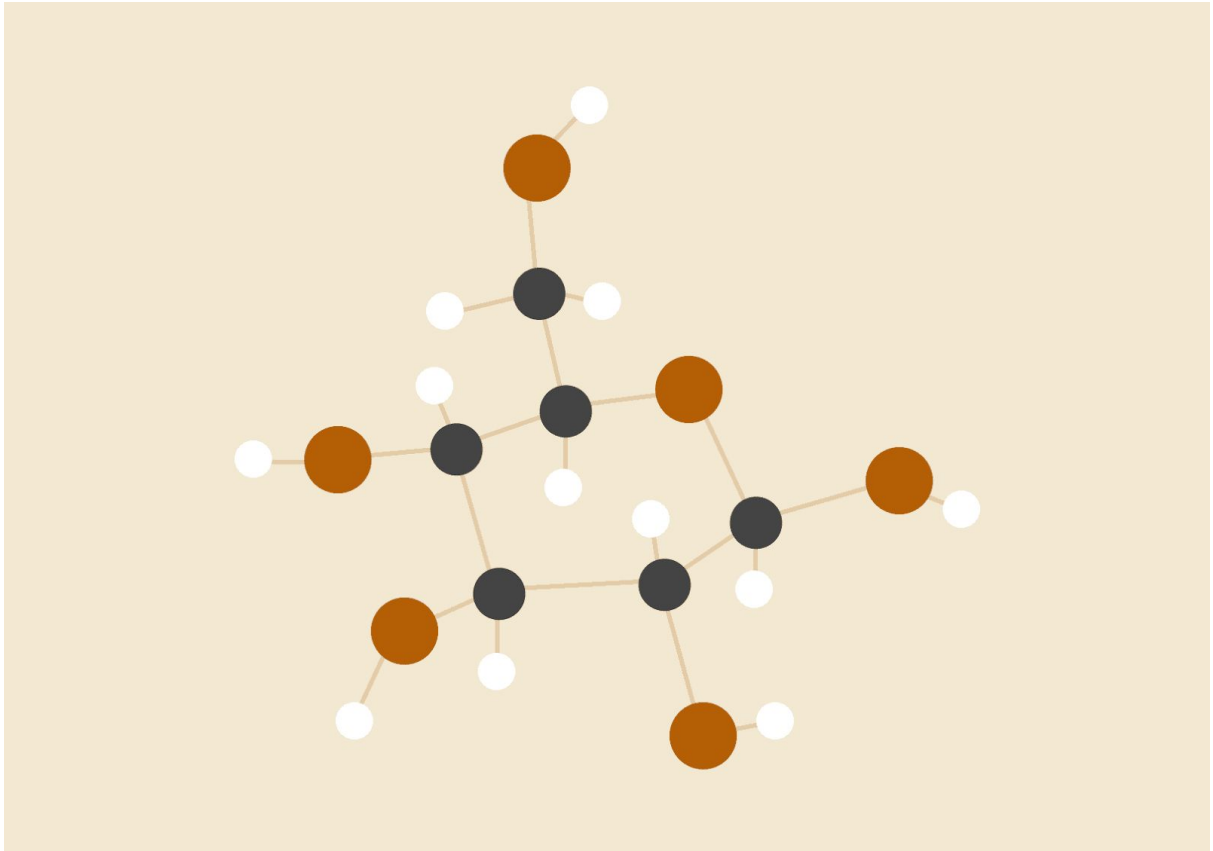


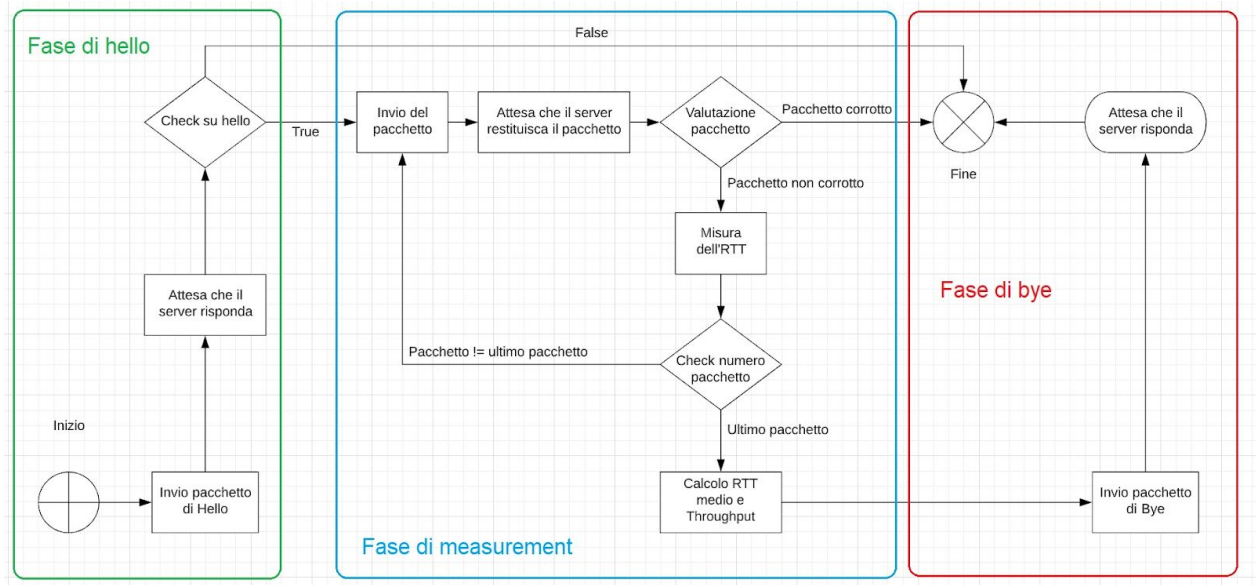
RELAZIONE ASSIGNMENT 3 DI RETI DI CALCOLATORI

Ismam Abu, Marco Hu, Zheliang Zhu



TASK 1:

Diagramma degli stati del Client:



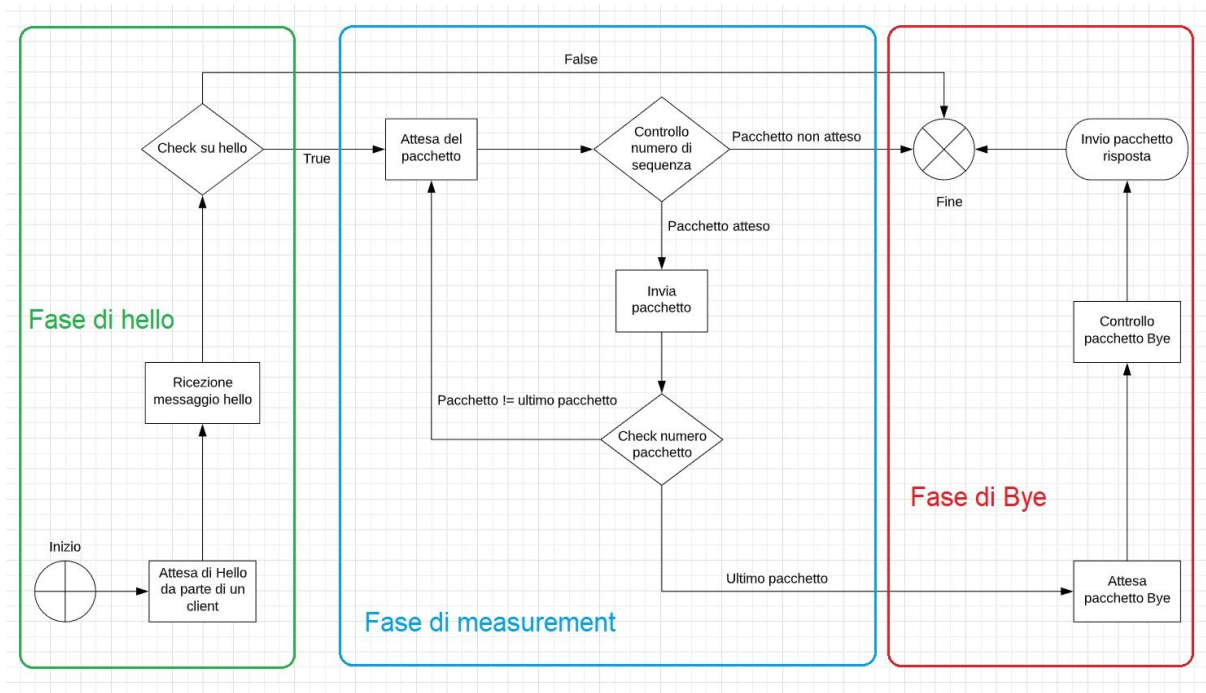
Il client per prima cosa effettua la fase di hello con il server per stabilire la connessione, questa viene stabilita internamente dal codice, l'utente non deve fornire nessun input.

Se la fase di hello non dovesse andare a buon fine si esce.

Completata la prima fase si passa al measurement: il client invia dei messaggi al server per eseguire le misure e calcolare l'RTT non appena riceve l'ultimo pacchetto inviato.

Calcolato RTT medio e Throughput si passa alla fase di Bye, il client manda il pacchetto di bye e attende che il server risponda per chiudere la connessione.

Diagramma degli stati del Server:



Il server inizialmente è in attesa che un client mandi un messaggio di hello.

Una volta ricevuto il messaggio, il server controlla quello che riceve, nel caso ricevesse qualcosa di errato conclude la connessione.

Se il messaggio è corretto si invia la conferma del suddetto ed il server passa alla ricezione dei pacchetti da parte del client entrando in fase di measurement.

In questa fase si controlla il numero di sequenza del pacchetto, in caso non fosse un pacchetto atteso si conclude la connessione, altrimenti si passa all'invio del pacchetto al client, e se non è l'ultimo pacchetto si rimette in attesa che il client invii i pacchetti restanti.

Se il pacchetto è l'ultimo il server va in fase di bye e resta in attesa di bye da parte del client, una volta ricevuto lo controlla, invia la risposta e chiude la connessione.

TASK 2:

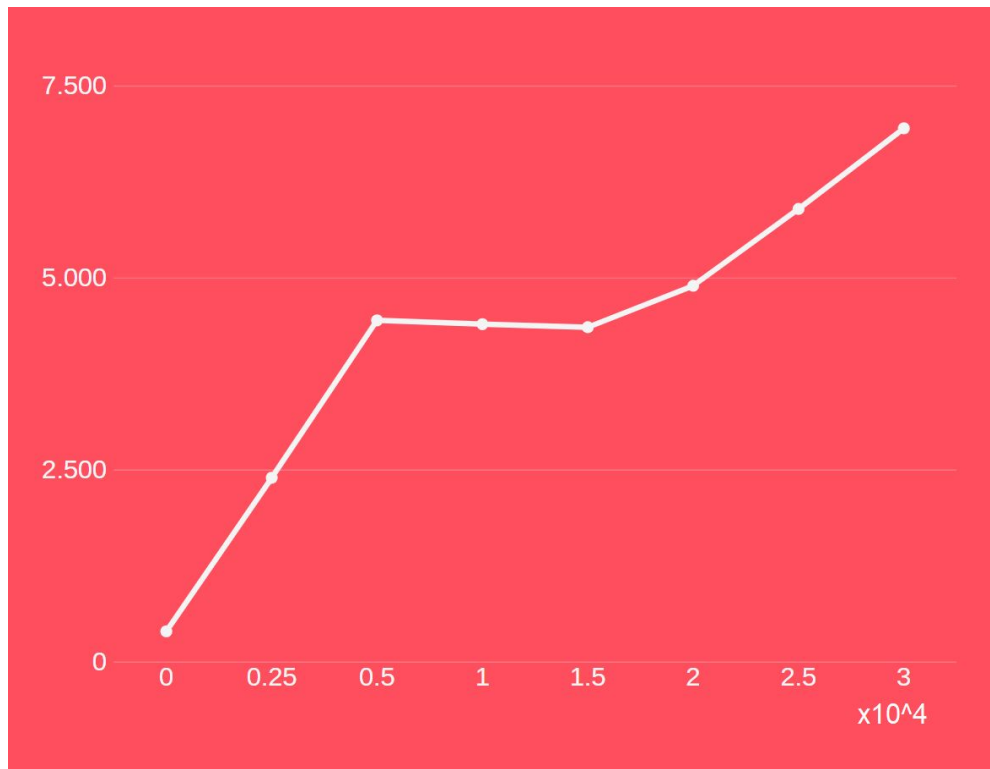
Strategia risolutiva:

La base è stato il codice fornito a lezione (TCP client e TCP server), da lì lo si è modificato in base alle nostre esigenze, sopra è stata implementata la fase di hello, il controllo della conferma del server.

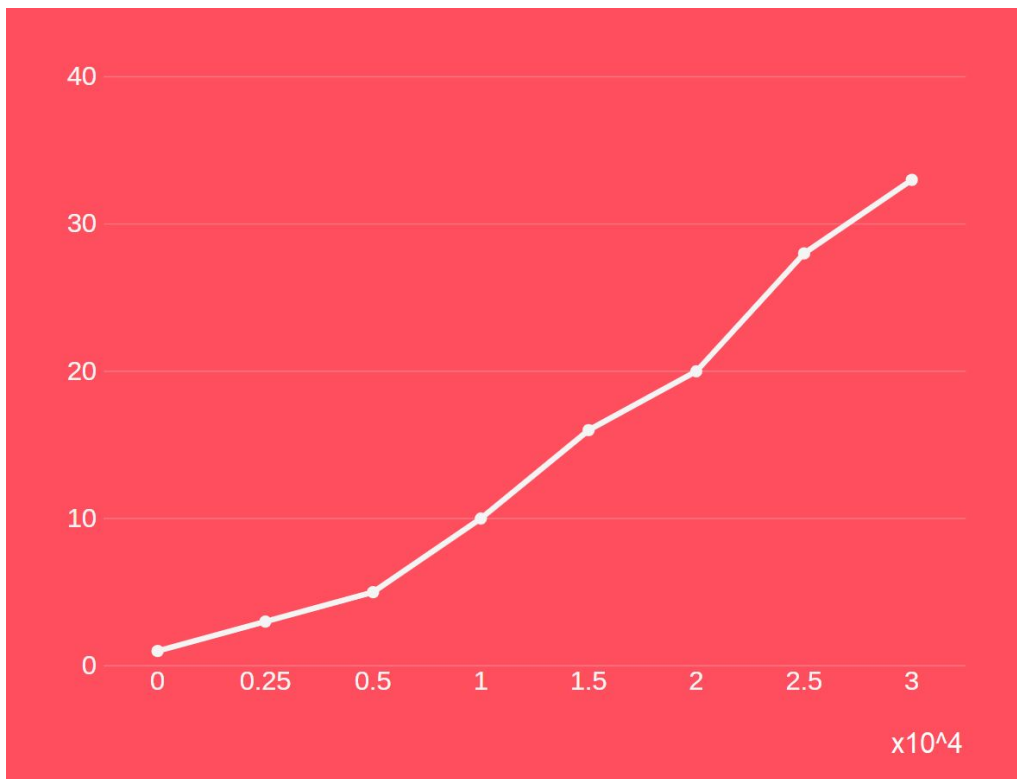
Successivamente si è passati alla fase di measurement, i vari parametri sono stati salvati nelle variabili (le nprobes, type per capire il tipo di pacchetto, payload).

In fine si passa alla fase di bye, il client manda la richiesta, il server manda la conferma e si chiude.

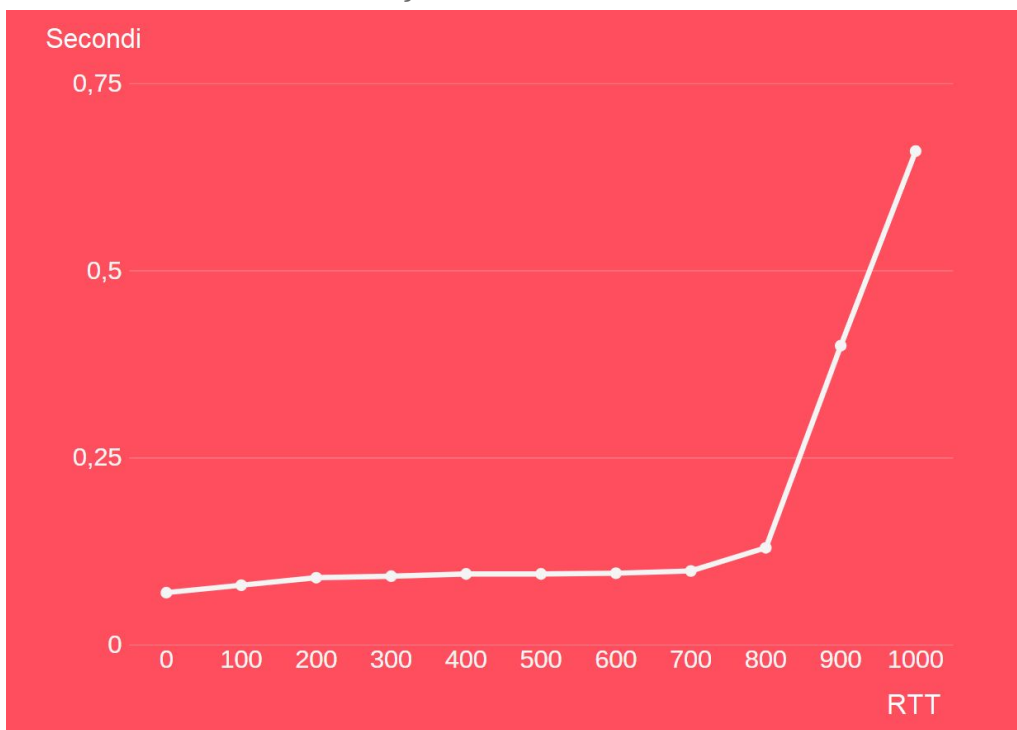
Throughput Medio senza server



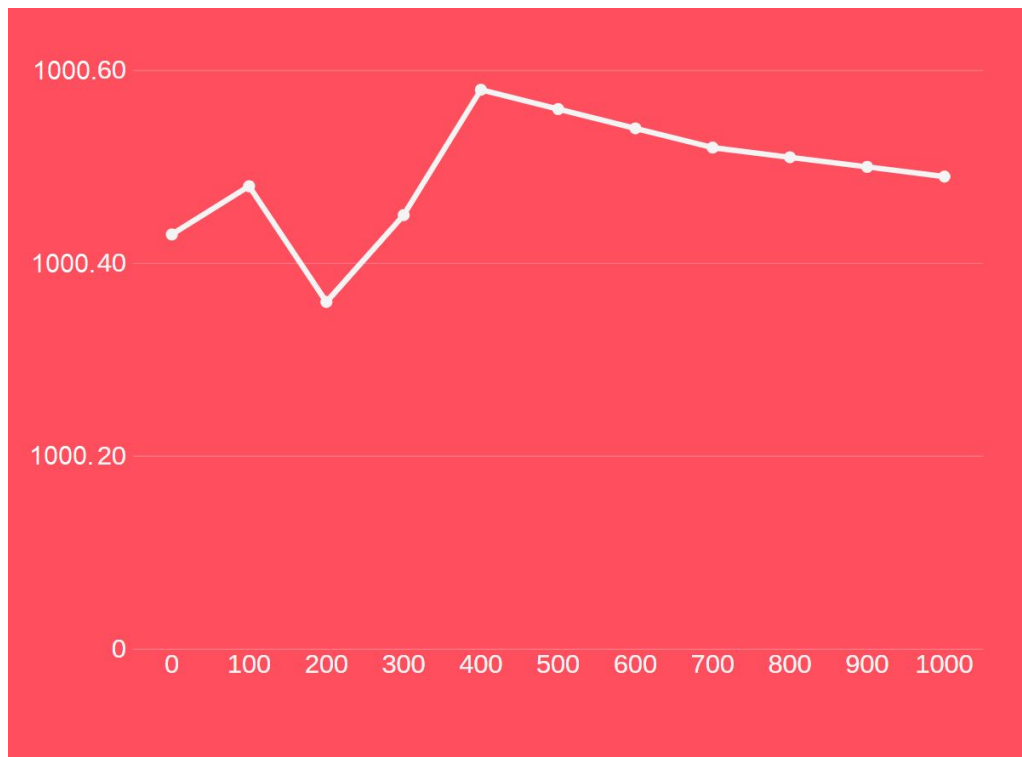
Throughput Medio con server delay di 1 secondo



RTT Medio senza server delay



RTT Medio con server delay di 1 secondo



Considerazioni sui grafici:

Nei due grafici relativi al throughput si nota che nel caso dove le misure sono state effettuate senza un server delay abbiamo che il throughput cresce al crescere delle dimensioni del pacchetto inviato, nel caso del server delay di 1 secondo si nota sempre una crescita all'aumentare dei byte inviati, è inferiore in assenza di server delay siccome l'RTT medio è maggiore.

Nei grafici relativi all'RTT, nel caso senza delay notiamo che l'RTT non superi mai il secondo ma cresca in un certo intervallo con l'aumentare della dimensione del pacchetto inviato

Nel caso con server delay invece i valori non scendono mai sotto il secondo.