Relazione

# Descrizione

Il progetto è stato sviluppato da Massaro Giuseppe Simone. Il software è in linguaggio C ed è implemento tramite le librerie di sistema POSIX.

Il progetto è costituito da un programma server ed un programma client, in cui il programma server calcola media e varianza di una serie di numeri interi positivi forniti dal client, secondo le specifiche sotto indicate.

Client

Il programma client viene eseguito indicando sulla riga di comando l’indirizzo IPv4 del server da contattare e la porta sulla quale contattarlo ($ programma\_client <indirizzo\_server> <numero\_porta>).

Il client elabora sempre i messaggi che riceve dal server, ovvero non ne presenta il contenuto direttamente all’utente, ma rimuove qualsiasi delimitatore del protocollo e mostra all’utente l’informazione ricevuta in maniera chiara e contestualizzata

Dopo l’apertura della connessione il client si aspetta di ricevere dal server il messaggio di benvenuto, nel formato OK START <Messaggio> ovvero la stringa ‘OK START’, seguita da uno spazio e da una stringa personalizzata dal server. Il messaggio deve avere una lunghezza massima di 512 caratteri ed è terminato da un carattere di andata a capo (‘\n’)

Il client spiega chiaramente all’utente lo scopo del programma, le opzioni a disposizione, il formato atteso e le modalità di funzionamento previsto.

Il client sollecita l’utente ad inserire i numeri di cui deve essere calcolata media e varianza, secondo il criterio che preferisce.

Una volta verificata la congruità dei dati inseriti, in relazione allo scopo del programma, il client trasmette i dati al server all’interno di uno o più messaggi.

Quando il client non ha più dati da trasmettere manda un messaggio con `0` (zero).

Dopo aver trasmesso ogni messaggio, il client si pone in attesa della risposta del server.

# Server

Il programma server viene eseguito indicando sulla riga di comando la porta sulla quale mettersi in ascolto ($ programma\_server <numero porta>).

I messaggi inviati dal server hanno una lunghezza massima di 512 caratteri, devono essere terminati da un carattere di andata a capo (‘\n’), e hanno il formato generale <Esito> <Tipo> <Contenuto>

* <Esito> identifica l’esito positivo o negativo del messaggio e può assumere i valori ‘OK’ ed ‘ERR’.
* <Tipo> identifica il comando al quale la risposta fa riferimento, o la categoria di risposta.
* <Contenuto> è il contenuto della risposta.

All’apertura della connessione il server manda un messaggio di benvenuto, poi il server si pone in attesa di un messaggio di comando da parte del client.

Il server esamina il messaggio ricevuto e ne verifica la correttezza sintattica. Se il messaggio non è sintatticamente corretto risponde con il messaggio di errore e chiude la connessione.

Se il numero di dati letti è corretto e maggiore di zero, memorizza i dati e risponde con il messaggio `OK DATA <numero\_dati\_letti>` ovvero la stringa `OK DATA` seguita da uno spazio e da una stringa numerica che rappresenta il valore numero di dati estratti dal messaggio ricevuto.

Se il numero di dati letti non è corretto, ovvero non il valore dichiarato non è coerente con il contenuto del messaggio, il server risponde con il messaggio `ERR DATA <messaggio>` ovvero la stringa `ERR DATA` seguita da uno spazio e da un messaggio di testo che descrive la natura dell’errore, e chiude la connessione.

Se il numero di dati letti è uguale a `0`, ovvero il client aveva mandato il messaggio `0` il server elabora tutti i dati ricevuti e ne calcola la media e la varianza dei campioni.

Dopo avere mandato il messaggio finale o aver mandato i messaggi di errore, il server chiude la connessione e si pone in attesa della richiesta di un nuovo client.