

# Programmazione multithread con i socket

<b>Esercizio A - Multi Threaded TcpServer.....</b>	<b>2</b>
Descrizione del Problema.....	2
Implementazione.....	2
Collaudo.....	2
→ Classe Thread.....	2
→ Interfaccia Runnable.....	4
→ Lambda.....	7
Procedimento e Risultati.....	9
Considerazioni finali.....	9
Approfondimento.....	9
<b>Esercizio B - Calcolatrice Remota.....</b>	<b>10</b>
Descrizione del Problema.....	10
Progettazione.....	10
Schema Logico.....	10
Implementazione.....	10
Collaudo.....	11
Procedimento e Risultati.....	12
Considerazioni Finali.....	13

# Esercizio A - Multi Threaded TcpServer

## Descrizione del Problema

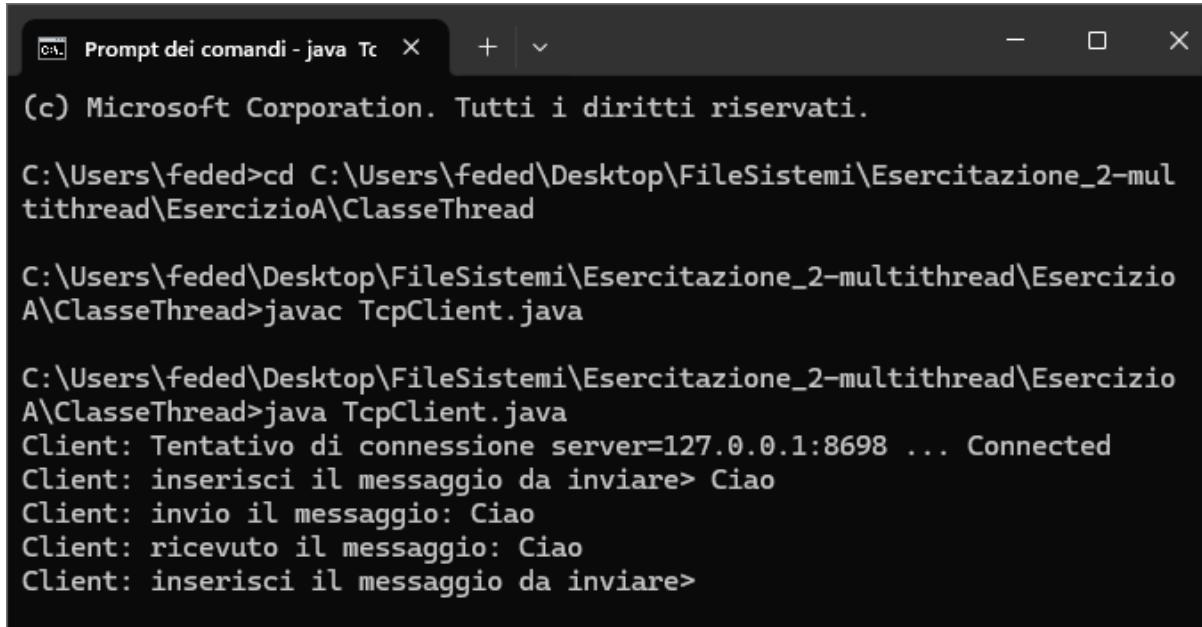
- Testare il funzionamento del codice di esempio multithread
- Archiviare il file nella propria cartella remota (es. percorso: /ES02a-MultiThreadedServer-runnable/)
- Consegnare un documento di Google contenente gli screenshot che ne dimostra il funzionamento, e il link al file di codice.

## Implementazione

- [esempio con Classe Thread](#)
- [esempio con runnable](#)
- [esempio con Lambda](#)

## Collaudo

→ **Classe Thread**



```
Prompt dei comandi - java Tc + | ▾ - □ ×
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>javac TcpClient.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>java TcpClient.java
Client: Tentativo di connessione server=127.0.0.1:8698 ... Connected
Client: inserisci il messaggio da inviare> Ciao
Client: invio il messaggio: Ciao
Client: ricevuto il messaggio: Ciao
Client: inserisci il messaggio da inviare>
```

```
Prompt dei comandi - java Tc  X  + | ▾  -  □  X

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>java TcpClient.java
Client: Tentativo di connessione server=127.0.0.1:8698 ... Connected
Client: inserisci il messaggio da inviare> Mondo
Client: invio il messaggio: Mondo
Client: ricevuto il messaggio: Mondo
Client: inserisci il messaggio da inviare>
```

```
Prompt dei comandi - java Tc  X  + | ▾  -  □  X

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>java TcpClient.java
Client: Tentativo di connessione server=127.0.0.1:8698 ... Connected
Client: inserisci il messaggio da inviare> Testing Esercizio A ClasseThread
Client: invio il messaggio: Testing Esercizio A ClasseThread
Client: ricevuto il messaggio: Testing Esercizio A ClasseThread
Client: inserisci il messaggio da inviare>
```

```
Prompt dei comandi - java M  X  + | ▾  -  □  X

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>javac Multithread

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\ClasseThread>java Multithread
Server: in ascolto sulla porta 8698
Serving Client 1
Server: in ascolto sulla porta 8698
Serving Client 2
Server: in ascolto sulla porta 8698
Serving Client 3
Server: in ascolto sulla porta 8698
Server.Thread 1 Ricevuto messaggio Ciao
Server.Thread 1 Invio messaggio Ciao
Server.Thread 2 Ricevuto messaggio Mondo
Server.Thread 2 Invio messaggio Mondo
Server.Thread 3 Ricevuto messaggio Testing Esercizio A ClasseThread
Server.Thread 3 Invio messaggio Testing Esercizio A ClasseThread
```

The image shows four terminal windows side-by-side, each running a different Java application.

- Top Left Terminal:** Shows the output of a Java client application named `TcpClient.java`. It connects to a server at port 8698 and exchanges messages with it.
- Top Right Terminal:** Shows the output of a Java server application named `MultithreadedTcpServer.java`. It handles multiple clients simultaneously, receiving "Mondo" and "Ciao" messages and sending "Mondo" and "Ciao" messages back.
- Bottom Left Terminal:** Shows the output of another Java client application named `TcpClient.java`, which also connects to the server and exchanges "Mondo" and "Ciao" messages.
- Bottom Right Terminal:** Shows the output of the Java server application again, this time with three clients connected and interacting.

## → Interfaccia Runnable

This terminal window displays the interaction between a Java client and a server using the `Runnable` interface.

```

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>javac TcpClient.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpClient.java
Connesso al server localhost:8080
Messaggio (o 'quit'): Ciao
Server: Echo: Ciao
Messaggio (o 'quit'): quit
  
```

```
Prompt dei comandi + - X Microsoft Windows [Versione 10.0.26200.7171] (c) Microsoft Corporation. Tutti i diritti riservati. C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpClient.java Connesso al server localhost:8080 Messaggio (o 'quit'): Mondo Server: Echo: Mondo Messaggio (o 'quit'): quit Client disconnesso. C:\Users\fedor\Desktop\FileSistemi\Esercitazione_2-multithread\Esercizio
```

```
Prompt dei comandi + - X Microsoft Windows [Versione 10.0.26200.7171] (c) Microsoft Corporation. Tutti i diritti riservati. C:\Users\fedor>cd C:\Users\fedor\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable C:\Users\fedor\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpClient.java Connesso al server localhost:8080 Messaggio (o 'quit'): Testing Esercizio A Runnable Server: Echo: Testing Esercizio A Runnable Messaggio (o 'quit'): quit Client disconnesso. C:\Users\fedor\Desktop\FileSistemi\Esercitazione_2-multithread\Esercizio
```

```
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>javac TcpServer.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpServer.java
Server TCP avviato sulla porta 8080
Client connesso: /127.0.0.1:59108
Client connesso: /127.0.0.1:59109
Client connesso: /127.0.0.1:62299
Ricevuto: Ciao
Ricevuto: Mondo
Ricevuto: Testing Esercizio A Runnable
```

The image displays four separate windows, each showing a command prompt (Prompt dei comandi) running on Microsoft Windows. The windows are arranged in a 2x2 grid.

- Top-left window:** Shows the server side. It starts with the command `javac TcpServer.java`, followed by the server starting message "Server TCP avviato sulla porta 8080". Then it lists three client connections with their respective messages: "Ciao", "Mondo", and "Testing Esercizio A Runnable".
- Top-right window:** Shows the client side. It starts with the command `java TcpClient.java`, followed by the connection message "Connesso al server localhost:8080". It then shows the server's echo of the client's message "Mondo", followed by the client's message "quit", and the server's response "Client disconnesso".
- Bottom-left window:** Shows the client side. It starts with the command `java TcpClient.java`, followed by the connection message "Connesso al server localhost:8080". It then shows the server's echo of the client's message "Testing Esercizio A Runnable", followed by the client's message "quit", and the server's response "Client disconnesso".
- Bottom-right window:** Shows the server side. It starts with the command `javac TcpServer.java`, followed by the server starting message "Server TCP avviato sulla porta 8080". Then it lists three client connections with their respective messages: "Ciao", "Mondo", and "Testing Esercizio A Runnable".

```
Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>javac TcpClient.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpClient.java
Connesso al server localhost:8080
Messaggio (o 'quit'): Mondo
Server: Echo: Mondo
Messaggio (o 'quit'): quit
Client disconnesso.

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>javac TcpServer.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\InterfacciaRunnable>java TcpServer.java
Server TCP avviato sulla porta 8080
Client connesso: /127.0.0.1:59108
Client connesso: /127.0.0.1:59109
Client connesso: /127.0.0.1:62299
Ricevuto: Ciao
Ricevuto: Mondo
Ricevuto: Testing Esercizio A Runnable
```

→ Lambda

```
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>javac TcpClient.java  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>java TcpClient.java  
Connesso al server localhost:8765  
Messaggio (o 'quit'): Ciao  
Server: Echo: Ciao  
Messaggio (o 'quit'): quit  
Client disconnesso.  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\Esercizio
```

```
Microsoft Windows [Versione 10.0.26200.7171]  
(c) Microsoft Corporation. Tutti i diritti riservati.  
C:\Users\feded>  
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>javac TcpClient.java  
Connesso al server localhost:8765  
Messaggio (o 'quit'): Mondo  
Server: Echo: Mondo  
Messaggio (o 'quit'): quit  
Client disconnesso.
```

```
Prompt dei comandi
Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda
bda

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>java TcpClient.java
Connesso al server localhost:8765
Messaggio (o 'quit'): Testing Esercizio A Lambda
Server: Echo: Testing Esercizio A Lambda
Messaggio (o 'quit'): quit
Client disconnesso.

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\Esercizio
```

```
Prompt dei comandi -java Tc
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda
bda

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>javac TcpServer.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioA\Lambda>java TcpServer.java
Server TCP avviato sulla porta 8765
Client connesso: /127.0.0.1:51656
Client connesso: /127.0.0.1:51657
Client connesso: /127.0.0.1:53533
Ricevuto: Ciao
Ricevuto: Mondo
Ricevuto: Testing Esercizio A Lambda
```

The image displays four separate terminal windows, each showing command-line output from a Windows environment. The windows are arranged in a 2x2 grid.

- Top-left window:** Shows the execution of a Java client named `TcpClient.java`. It connects to a server at localhost:8765 and exchanges messages with it. The client sends "Ciao" and receives "Echo: Ciao". It then sends "quit" and receives "Mondo", followed by a disconnection message.
- Top-right window:** Shows the execution of a Java client named `TcpClient.java`. It connects to a server at localhost:8765 and exchanges messages with it. The client sends "quit" and receives "Mondo", followed by a disconnection message.
- Bottom-left window:** Shows the execution of a Java client named `TcpClient.java`. It connects to a server at localhost:8765 and exchanges messages with it. The client sends "quit" and receives "Mondo", followed by a disconnection message.
- Bottom-right window:** Shows the execution of a Java server named `TcpServer.java`. It starts listening on port 8765, connects to a client at 127.0.0.1:51656, and exchanges messages with it. The server receives "Ciao" and replies with "Ricevuto: Ciao". It then receives "Mondo" and replies with "Ricevuto: Mondo". Finally, it receives "Testing Esercizio A Lambda" and replies with "Ricevuto: Testing Esercizio A Lambda".

## Procedimento e Risultati

Il testing del server multithread è stato effettuato seguendo diverse fasi di verifica.

Inizialmente è stata testata la connessione con un singolo client, utilizzando un semplice client TCP. Il server ha risposto correttamente a tutti i messaggi inviati, dimostrando il corretto funzionamento della comunicazione base.

Successivamente è stata verificata la gestione di più client simultanei. Tre diversi client sono stati connessi contemporaneamente al server, inviando messaggi in sequenza. Il sistema ha gestito correttamente tutte le connessioni, mantenendo separati i dialoghi e rispondendo in modo appropriato a ciascun client.

## Considerazioni finali

Il codice fornito ha dimostrato piena funzionalità nelle condizioni di test applicate. Il server multithread è in grado di gestire correttamente più connessioni parallele, con ogni thread che opera isolato dagli altri.

L'architettura multithread si conferma adatta per applicazioni server che richiedono la gestione di più client simultanei.

## Approfondimento

Per ottimizzare le prestazioni in caso di picchi di carico elevati, sarebbe opportuno considerare l'implementazione di un meccanismo di thread pooling, ossia una raccolta di thread riutilizzabili che vengono mantenuti attivi per eseguire operazioni su richiesta.

# Esercizio B - Calcolatrice Remota

## Descrizione del Problema

- Modificare l'esercitazione precedente “[calcolatrice](#)” per accettare connessioni da più client contemporaneamente  
Aggiungere nei messaggi visualizzati a schermo il nome del server (OPZIONALE) aggiungere anche il nome e/o indirizzo ip del client
- Archiviare il file nella propria cartella remota (es. percorso: /ES02b-CalcolatriceMultithread/)
- Consegnare un documento di Google contenente gli screenshot che ne dimostra il funzionamento, e il link al file di codice.
- [CalcolatriceClient](#) precedente
- [CalcolatriceServer](#) precedente

## Progettazione

La modifica principale riguarda la struttura del server, che deve diventare multithread. Per ogni nuovo client in ingresso, il server crea un nuovo thread dedicato alla gestione della connessione, in modo simile all'esempio multithread con Runnable utilizzato nell'Esercizio A.

### Schema Logico

- Il server apre la ServerSocket e rimane in ascolto.
- Ogni volta che un client si connette, viene creato un thread che gestisce:
  - la lettura delle operazioni;
  - l'elaborazione (somma, sottrazione, moltiplicazione, divisione);
  - l'invio del risultato;
- Il thread rimane attivo finché il client non invia un comando di uscita.
- Il server principale rimane libero di accettare nuove connessioni.

## Implementazione

- [CalcolatriceClient](#)
- [CalcolatriceServer](#)

## Collaudo

```
C:\ Prompt dei comandi + - ×  
tithread\EsercizioB  
  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>javac CalcolatriceClient.java  
  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>java CalcolatriceClient.java  
CALCOLATRICE REMOTA  
Formato: NUMERO OPERAZIONE NUMERO  
Operazioni supportate: + - * /  
Scrivi 'quit' per uscire.  
Calcolo> 2 * 1938  
Server: 3876.0  
Calcolo> quit  
Server: Arrivederci
```

```
C:\ Prompt dei comandi + - ×  
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-mul  
tithread\EsercizioB  
  
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>javac CalcolatriceClient.java  
CALCOLATRICE REMOTA  
Formato: NUMERO OPERAZIONE NUMERO  
Operazioni supportate: + - * /  
Scrivi 'quit' per uscire.  
Calcolo> 2 + 10  
Server: 12.0  
Calcolo> 10 / 10  
Server: 1.0  
Calcolo> quit  
Server: Arrivederci
```

```
Prompt dei comandi - java Ci X + | - □ ×

Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>javac CalcolatriceServer.java

C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>java CalcolatriceServer.java
Server avviato sulla porta 8844
Nuova connessione accettata
Nuova connessione accettata
Richiesta ricevuta: 2 + 10
Richiesta ricevuta: 2 * 1938
Richiesta ricevuta: QUIT
Richiesta ricevuta: 10 / 10
Richiesta ricevuta: QUIT
```

```
Prompt dei comandi - java Ci X + | - □ ×
tithread\EsercizioB
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>javac CalcolatriceClient.java
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>java CalcolatriceClient.java
CALCOLATRICE REMOTA
Formato: NUMERO OPERAZIONE NUMERO
Operazioni supportate: + - * /
Scrivi 'quit' per uscire.
Calcolo> 2 * 1938
Server: 3876.0
Calcolo> quit
Server: Arrivederci

Prompt dei comandi - java Ci X + | - □ ×
C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>javac CalcolatriceServer.java
Microsoft Windows [Versione 10.0.26200.7171]
(c) Microsoft Corporation. Tutti i diritti riservati.

C:\Users\feded>cd C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB
C:\Users\feded\Desktop\FileSistemi\Esercitazione_2-multithread\EsercizioB>java CalcolatriceServer.java
Server avviato sulla porta 8844
Nuova connessione accettata
Nuova connessione accettata
Richiesta ricevuta: 2 + 10
Richiesta ricevuta: 2 * 1938
Richiesta ricevuta: QUIT
Richiesta ricevuta: 10 / 10
Richiesta ricevuta: QUIT
```

## Procedimento e Risultati

Per verificare il corretto funzionamento della calcolatrice multithread, sono stati effettuati diversi test.

Durante i primi test è emerso un problema: la connessione veniva chiusa subito dopo il primo calcolo. L'errore era dovuto al BufferedReader che non era inserito in un ciclo continuo. La soluzione è stata quella di mantenere aperta la connessione e leggere più righe tramite un ciclo while, permettendo al client di inviare operazioni finché non decide di terminare.

Successivamente sono stati testati più client in parallelo. L'esecuzione simultanea ha confermato che ogni client viene gestito da un thread separato: le operazioni non interferiscono tra loro e il server registra correttamente le richieste.

### **Considerazioni Finali**

Lo sviluppo consisteva soprattutto nell'adattare il server alla struttura multithread, creando un gestore dedicato per ogni nuova connessione. Una volta risolto il problema del ciclo di lettura, il sistema ha funzionato senza ulteriori modifiche.

In conclusione, la calcolatrice aggiornata gestisce correttamente più connessioni contemporanee e permette a ogni client di inviare più operazioni consecutive. L'applicazione risulta stabile e reattiva, e rappresenta un buon esempio di come il multithreading migliori la gestione delle comunicazioni in un server TCP.