

INIZIO ServerMultiThread

porta ← 12345

serverSocket ← crea ServerSocket sulla porta

STAMPA "Server avviato sulla porta", porta

STAMPA "In attesa di connessioni..."

MENTRE vero FARE

clientSocket ← serverSocket accetta()

STAMPA "Nuovo client connesso:", indirizzo del clientSocket

clientHandler ← crea nuovo ClientHandler(clientSocket)

thread ← crea nuovo Thread(clientHandler)

thread.avvia()

STAMPA "Thread avviato per client:", indirizzo del clientSocket

FINE MENTRE

FINE ServerMultiThread

---

INIZIO ClientHandler

VARIABILI:

socket: Socket del client

COSTRUTTORE(socket)

this.socket ← socket

FINE COSTRUTTORE

METODO run()

PROVA

in ← crea BufferedReader dal socket.getInputStream()

out ← crea PrintWriter dal socket.getOutputStream()

MENTRE (inputLine ← in.leggiLinea()) != null FARE

SE inputLine = "BYE" (ignora maiuscole/minuscole) ALLORA

out.stampa("Arrivederci!")

ESCI dal ciclo

FINE SE

risposta ← calcolaPoligono(inputLine)

out.stampa(risposta)

FINE MENTRE

CATTURA IOException

stampa errore

FINALMENTE

chiudi socket

FINE PROVA  
FINE METODO

METODO calcolaPoligono(input)

PROVA

parti ← dividi input per spazi

SE lunghezza(parti) ≠ 2 ALLORA

RITORNA "ERRORE: Formato non valido"

FINE SE

numeroLati ← converti parti[0] a intero

lunghezzaLato ← converti parti[1] a double

SE numeroLati < 3 ALLORA

RITORNA "ERRORE: Un poligono deve avere almeno 3 lati"

FINE SE

SE lunghezzaLato ≤ 0 ALLORA

RITORNA "ERRORE: La lunghezza del lato deve essere positiva"

FINE SE

perimetro ← numeroLati × lunghezzaLato

area ← (numeroLati × lunghezzaLato<sup>2</sup>) / (4 × tan( $\pi$  / numeroLati))

RITORNA formatta risultato con lati, lunghezza, perimetro, area

CATTURA NumberFormatException

RITORNA "ERRORE: I valori devono essere numerici"

CATTURA Exception

RITORNA "ERRORE: " + messaggio errore

FINE PROVA

FINE METODO

FINE ClientHandler

---

INIZIO ClienteSemplice

hostname ← "127.0.0.1"

porta ← 12345

PROVA

socket ← crea Socket(hostname, porta)

out ← crea PrintWriter dal socket.getOutputStream()

in ← crea BufferedReader dal socket.getInputStream()

scanner ← crea Scanner per System.in

```
STAMPA "=== CALCOLATORE POLIGONI REGOLARI ==="
STAMPA "Connesso al server."
STAMPA "Inserisci: numeroLati lunghezzaLato"
STAMPA "Esempio: 5 10 (per un pentagono con lato di 10)"
STAMPA "Scrivi 'BYE' per uscire"

MENTRE scanner ha prossima linea FARE
    STAMPA "Input: "
    inputUtente ← scanner.leggiLinea()

    out.stampa(inputUtente)

    SE inputUtente = "BYE" (ignora maiuscole/minuscole) ALLORA
        STAMPA in.leggiLinea()
        ESCI dal ciclo
    FINE SE

    risposta ← in.leggiLinea()
    STAMPA risposta
FINE MENTRE

chiudi scanner

CATTURA IOException
    stampa errore
FINE PROVA
FINE ClienteSemplice
```