

Esercizi di Programmazione in C

Questa raccolta di esercizi copre vari argomenti della programmazione in linguaggio C, dal livello base alla comunicazione tra processi (IPC). Al termine sono inclusi alcuni esercizi facoltativi più complessi per chi desidera mettersi alla prova.

Livello base

1. Somma di un array: Scrivi un programma in C che calcoli la somma di tutti gli elementi di un array di interi.
2. Ricerca di un elemento: Scrivi un programma che verifichi se un numero dato è presente in un array.
3. Fattoriale ricorsivo e iterativo: Implementa due funzioni per calcolare il fattoriale di un numero, una ricorsiva e una iterativa.

Livello intermedio

1. Inverti una stringa: Scrivi una funzione che inverta una stringa senza usare funzioni di libreria specifiche.
2. Calcolo del massimo e minimo di un array: Scrivi un programma che trovi il valore massimo e minimo all'interno di un array di interi.
3. Conta vocali e consonanti: Dato un testo inserito dall'utente, conta il numero di vocali e consonanti presenti.

Comunicazione tra processi (IPC)

1. Fork e stampa dei PID: Scrivi un programma che utilizzi `fork()` per creare un processo figlio. Il padre e il figlio devono stampare il proprio PID e il PID del processo genitore.
2. Pipe tra padre e figlio: Crea una pipe. Fai il `fork()`. Il padre scrive una stringa nella pipe. Il figlio legge la stringa dalla pipe e la stampa.
3. Uso dei segnali: Scrivi un programma che imposti un gestore per il segnale `SIGUSR1`. Il padre invia un segnale `SIGUSR1` al figlio dopo 3 secondi dalla creazione. Il figlio stampa un messaggio di conferma.
4. Memoria condivisa: Scrivi un programma che utilizzi la memoria condivisa (`shmget`, `shmat`, `shmdt`, `shmctl`) per permettere a due processi di scambiarsi un messaggio.

Extra facoltativi (più avanzati)

1. Somma parallela di array con fork: Dividi un array di numeri interi tra più processi figli tramite `fork()` e pipe. Ogni processo calcola la somma della propria parte e invia il risultato al padre. Il padre somma i risultati parziali.
2. Produttore-Consumatore con pipe: Implementa il problema produttore-consumatore usando due processi e una pipe. Il produttore genera numeri casuali e li scrive nella pipe, il consumatore li legge e li stampa.
3. Processo che si riavvia: Scrivi un programma in cui un processo padre crea un figlio. Quando il figlio termina, il padre crea immediatamente un nuovo figlio e così via, finché non si raggiunge un certo numero di figli generati.