Write

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <string.h>

#define SHM_SIZE 1024   // Size of shared memory
#define SHM_KEY 1234    // Unique key for shared memory

int main() {
    int shmid;
    char *shmaddr;

    // Create shared memory segment
    shmid = shmget(SHM_KEY, SHM_SIZE, IPC_CREAT | 0666);
    if (shmid < 0) {
        perror("shmget error");
        exit(1);
    }

    // Attach the shared memory segment
    shmaddr = (char *) shmat(shmid, NULL, 0);
    if (shmaddr == (char *) -1) {
        perror("shmat error");
        exit(1);
    }

    // Write message to shared memory
    printf("Enter a message to write into shared memory: ");
    fgets(shmaddr, SHM_SIZE, stdin);

    printf("Message written to shared memory successfully.\n");

    // Detach from shared memory
    shmdt(shmaddr);

    return 0;
}
```

READ

```
#include <stdio.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>

#define SHM_SIZE 1024
#define SHM_KEY 1234

int main() {
    int shmid;
    char *shmaddr;

    // Locate the shared memory segment created by the server
    shmid = shmget(SHM_KEY, SHM_SIZE, 0666);
    if (shmid < 0) {
        perror("shmget error");
        exit(1);
    }

    // Attach to the shared memory
    shmaddr = (char *) shmat(shmid, NULL, 0);
    if (shmaddr == (char *) -1) {
        perror("shmat error");
        exit(1);
    }

    // Read the message from shared memory
    printf("Message read from shared memory: %s\n", shmaddr);

    // Detach from shared memory
    shmdt(shmaddr);

    // Remove the shared memory segment after use
    shmctl(shmid, IPC_RMID, NULL);

    return 0;
}
```