

```

<html><head></head><body><pre style="word-wrap: break-word; white-space: pre-wrap;">/**
 * Simple program demonstrating shared memory in POSIX systems.
 *
 * This is the consumer process
 *
 * Figure 3.18
 *
 * @author Gagne, Galvin, Silberschatz
 * Operating System Concepts - Ninth Edition
 * Copyright John Wiley & Sons - 2013
 */

#include <stdio.h>;
#include <stdlib.h>;
#include <fcntl.h>;
#include <sys/shm.h>;
#include <sys/stat.h>;
#include <sys/mman.h>;

int main()
{
    const char *name = "OS";
    const int SIZE = 4096;

    int shm_fd;
    void *ptr;
    int i;

    /* open the shared memory segment */
    shm_fd = shm_open(name, O_RDONLY, 0666);
    if (shm_fd == -1) {
        printf("shared memory failed\n");
        exit(-1);
    }

    /* now map the shared memory segment in the address space of the process */
    ptr = mmap(0, SIZE, PROT_READ, MAP_SHARED, shm_fd, 0);
    if (ptr == MAP_FAILED) {
        printf("Map failed\n");
        exit(-1);
    }

    /* now read from the shared memory region */
    printf("%s", ptr);

    /* remove the shared memory segment */
    if (shm_unlink(name) == -1) {
        printf("Error removing %s\n", name);
        exit(-1);
    }

    return 0;
}
</pre></body></html>

```