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

#define FOO 4096

int main () {
    int shmId;
    //int kernel_index;
    //int shm_size;
    struct shmid_ds shm_data;
    size_t size_of_shm_segment;
    char *shmPtr;

    if ((shmId =
        shmget (IPC_PRIVATE, FOO,
            IPC_CREAT | S_IRUSR | S_IWUSR)) < 0) {
        perror ("i can't get no..\n");
        exit (1);
    }
    if ((shmPtr = shmat (shmId, 0, 0)) == (void *) -1) {
        perror ("can't attach\n");
        exit (1);
    }
    printf ("value a: %lu\t value b: %lu\n", (unsigned long) shmPtr,
            (unsigned long) shmPtr + FOO);
    if (shmdt (shmPtr) < 0) {
        perror ("just can't let go\n");
        exit (1);
    }

 // this works as well.
 //   kernel_index = shmctl(shmId, SHM_INFO, &shm_data);
 //   shm_size = shmctl(kernel_index, SHM_STAT, &shm_data);

    // get info on shmem_location
    shmctl(shmId, IPC_STAT, &shm_data);

    // printing shmem info
    size_of_shm_segment = shm_data.shm_segsz;
    printf("Size of segment (bytes) %g\n", (double) size_of_shm_segment);
    //printf("Kernel index: %c\n", kernel_index);

    if (shmctl (shmId, IPC_RMID, 0) < 0) {
        perror ("can't deallocate\n");
        exit (1);
    }

    return 0;
}
```