```
/* Filename: shm_write.c */

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

#define BUF_SIZE 1024
#define SHM_KEY 0x1234

struct shmseg {
   int cnt;
   int complete;
   char buf[BUF_SIZE];
};

int fill_buffer(char *bufptr, int size);

int main() {
   int shmid;
   struct shmseg *shmp;
   int numtimes;

   // Create shared memory segment
```

```
shmid = shmget(SHM_KEY, sizeof(struct shmseg), 0644 | IPC_CREAT);
if (shmid == -1) {
   perror("Error creating shared memory");
   exit(EXIT_FAILURE);
}

// Attach to the shared memory
shmp = shmat(shmid, NULL, 0);
if (shmp == (void *) -1) {
   perror("Error attaching shared memory");
   exit(EXIT_FAILURE);
}

printf("Writer: Attached to shared memory.\n");

for (numtimes = 0; numtimes < 5; numtimes++) {
   shmp->complete = 0; // Mark as not complete before writing
   shmp->cnt = fill_buffer(shmp->buf, BUF_SIZE);

   printf("Writer: Wrote %d bytes to shared memory.\n", shmp->cnt);

   sleep(3); // Simulate delay between writes
}

shmp->complete = 1; // Mark writing as complete
printf("Writer: Completed writing %d times.\n", numtimes);
```

```
    // Detach from shared memory
    if (shmdt(shmp) == -1) {
        perror("Error detaching shared memory");
        exit(EXIT_FAILURE);
    }

    // Delete the shared memory segment
    if (shmctl(shmid, IPC_RMID, NULL) == -1) {
        perror("Error removing shared memory");
        exit(EXIT_FAILURE);
    }

    printf("Writer: Shared memory detached and destroyed. Exiting.\n");
    return 0;
}

int fill_buffer(char *bufptr, int size) {
    static char ch = 'A';
    memset(bufptr, ch, size - 1);
    bufptr[size - 1] = '\0'; // Null terminate
    int filled_count = strlen(bufptr);

    // Loop character from A-Z then a-z
    if (ch >= 122) {
        ch = 'A';
    } else {
        if ((ch >= 'A' && ch <= 'Z') || (ch >= 'a' && ch <= 'z')) {
```

```
            ch++;
        } else {
            ch = 'A';
        }
    }

    return filled_count;
}

/*OUTPUT –
Writer: Attached to shared memory.
Writer: Wrote 1023 bytes to shared memory.
Writer: Wrote 1023 bytes to shared memory.
Writer: Wrote 1023 bytes to shared memory.
Writer: Wrote 1023 bytes to shared memory.
Writer: Wrote 1023 bytes to shared memory.
Writer: Completed writing 5 times.
Writer: Shared memory detached and destroyed. Exiting.
*/
```