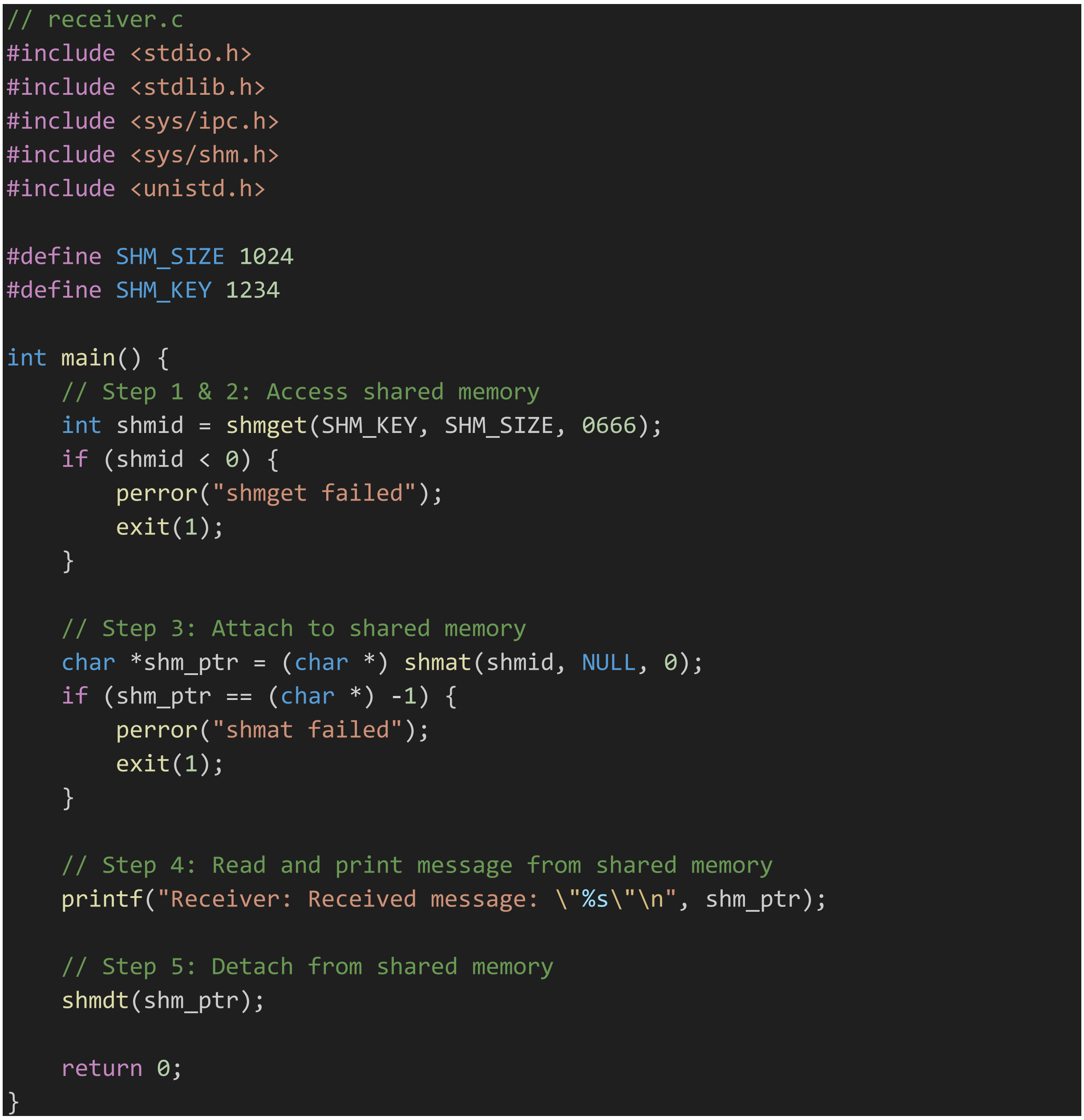
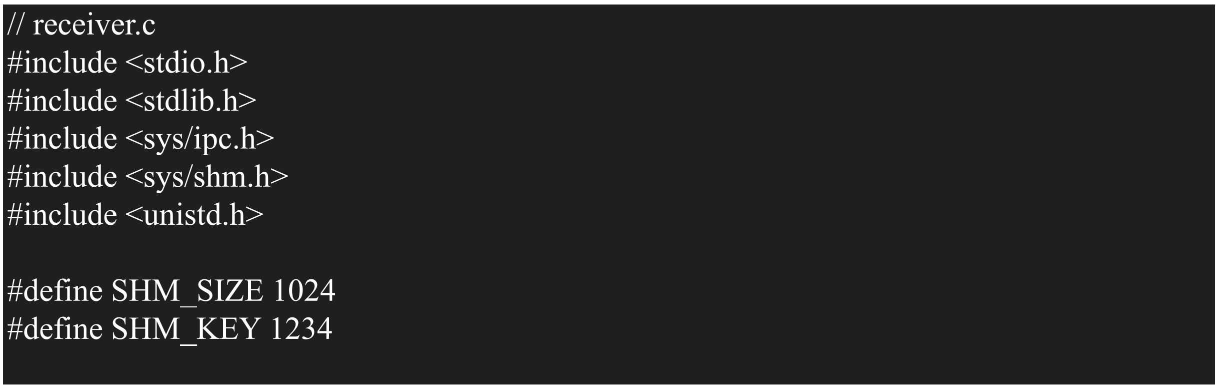
Expr 7: Sender and Receiver code in the Fedora

Sender code:



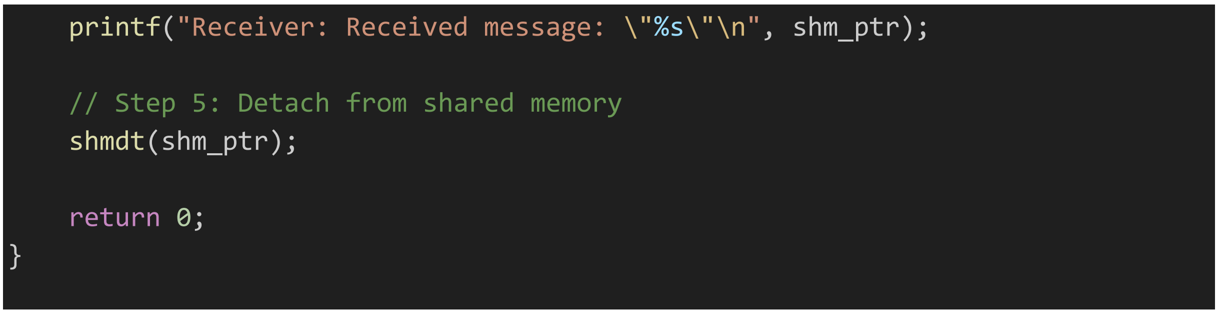
```c
// receiver.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <unistd.h>

#define SHM_SIZE 1024
#define SHM_KEY 1234

int main() {
    // Step 1 & 2: Access shared memory
    int shmid = shmget(SHM_KEY, SHM_SIZE, 0666);
    if (shmid < 0) {
        perror("shmget failed");
        exit(1);
    }

    // Step 3: Attach to shared memory
    char *shm_ptr = (char *) shmat(shmid, NULL, 0);
    if (shm_ptr == (char *) -1) {
        perror("shmat failed");
        exit(1);
    }

    // Step 4: Read and print message from shared memory
    printf("Receiver: Received message: \"%s\"\n", shm_ptr);

    // Step 5: Detach from shared memory
    shmdt(shm_ptr);

    return 0;
}
```

Receiver code:



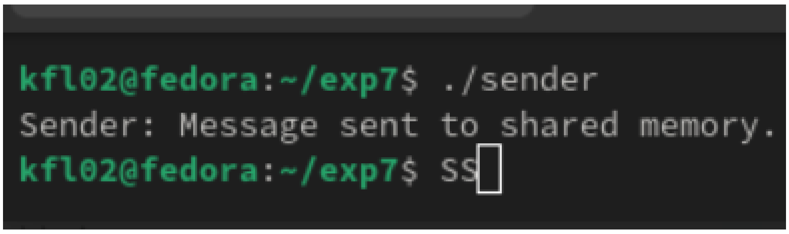
```c
// receiver.c
#include <stdio.h>
#include <stdlib.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#include <unistd.h>

#define SHM_SIZE 1024
#define SHM_KEY 1234
```

```
int main() {
   // Step 1 & 2: Access shared memory
   int shmid = shmget(SHM_KEY, SHM_SIZE, 0666);
   if (shmid < 0) {
      perror("shmget failed");
      exit(1);
   }

   // Step 3: Attach to shared memory
   char *shm_ptr = (char *) shmat(shmid, NULL, 0);
   if (shm_ptr == (char *) -1) {
      perror("shmat failed");
      exit(1);
   }

   // Step 4: Read and print message from shared memory
   printf("Receiver: Received message: \"%s\"\n", shm_ptr);

   // Step 5: Detach from shared memory
   shmdt(shm_ptr);

   return 0;
}// receiver.c

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

#define SHM_SIZE 1024
#define SHM_KEY 1234

int main() {
    // Step 1 & 2: Access shared memory
    int shmid = shmget(SHM_KEY, SHM_SIZE, 0666);
    if (shmid < 0) {
        perror("shmget failed");
        exit(1);
    }

    // Step 3: Attach to shared memory
    char *shm_ptr = (char *) shmat(shmid, NULL, 0);
    if (shm_ptr == (char *) -1) {
        perror("shmat failed");
        exit(1);
    }

    // Step 4: Read and print message from shared memory
```



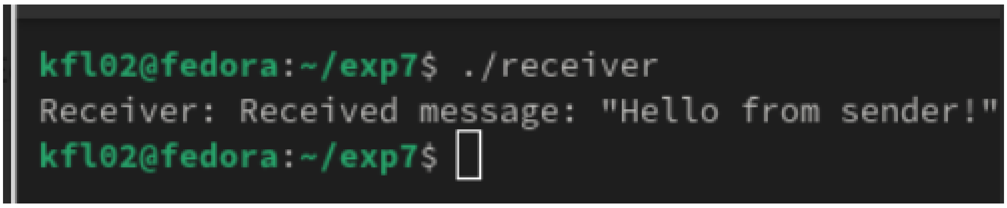
```
    printf("Receiver: Received message: \"%s\"\n", shm_ptr);

    // Step 5: Detach from shared memory
    shmdt(shm_ptr);

    return 0;
}
```

Sender Output:



```
kfl02@fedora:~/exp7$ ./sender
Sender: Message sent to shared memory.
kfl02@fedora:~/exp7$ SS
```

Receiver Output:



```
kfl02@fedora:~/exp7$ ./receiver
Receiver: Received message: "Hello from sender!"
kfl02@fedora:~/exp7$
```

Result:

Thus the Sender and Receiver Code is implemented in fedora using the C language