

LAB 6

Érica Oliveira Regnier - 2211893

Hanna Epelboim Assunção - 2310289

1- a)

Shared.h (cabeçalho)

```
#ifndef SHARED_H
#define SHARED_H

#include <semaphore.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>
#define MAX_MSG_LENGTH 16
#define NUM_MESSAGES 128

// Estrutura para a memória compartilhada
typedef struct {
    char message[MAX_MSG_LENGTH];
    int message_number;
} SharedMemory;

#define SHM_KEY 1234 // Chave para a memória compartilhada
#define SEM_WRITER_KEY 5678 // Chave para o semáforo do escritor
#define SEM_READER_KEY 91011 // Chave para o semáforo do leitor

#endif // SHARED_H
```

writer.c:

```
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <fcntl.h>
#include <sys/mman.h>
#include <semaphore.h>
#include <unistd.h>
#include "shared.h"
```

```

int main() {
// Criação de semáforos
sem_t *sem_writer = sem_open("/sem_writer", O_CREAT, 0666, 1);
sem_t *sem_reader = sem_open("/sem_reader", O_CREAT, 0666, 0);

// Criação da memória compartilhada
int shm_id = shmget(SHM_KEY, sizeof(SharedMemory), IPC_CREAT | 0666);
SharedMemory *shared_mem = (SharedMemory *)shmat(shm_id, NULL, 0);

for (int i = 1; i <= NUM_MESSAGES; i++) {
// Espera o semáforo do escritor estar disponível
sem_wait(sem_writer);

// Escreve a mensagem na memória compartilhada
snprintf(shared_mem->message, MAX_MSG_LENGTH, "mensagem %d", i);
shared_mem->message_number = i;
printf("Escritor: %s\n", shared_mem->message);

// Sinaliza para o processo leitor que a mensagem está pronta
sem_post(sem_reader);
}

// Limpeza
sem_close(sem_writer);
sem_close(sem_reader);
shmdt(shared_mem);
sem_unlink("/sem_writer");
sem_unlink("/sem_reader");
return 0;
}

```

reader.c:

```

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <fcntl.h>
#include <sys/mman.h>
#include <semaphore.h>
#include <unistd.h>
#include "shared.h"

int main() {

```

```

// Abertura de semáforos
sem_t *sem_writer = sem_open("/sem_writer", 0);
sem_t *sem_reader = sem_open("/sem_reader", 0);

// Acesso à memória compartilhada
int shm_id = shmget(SHM_KEY, sizeof(SharedMemory), 0666);
SharedMemory *shared_mem = (SharedMemory *)shmat(shm_id, NULL, 0);

for (int i = 1; i <= NUM_MESSAGES; i++) {
// Espera o semáforo do leitor
sem_wait(sem_reader);

// Lê a mensagem da memória compartilhada
printf("Leitor: %s\n", shared_mem->message);

// Sinaliza para o escritor que a mensagem foi lida
sem_post(sem_writer);
}

// Limpeza
sem_close(sem_writer);
sem_close(sem_reader);
shmdt(shared_mem);
return 0;
}

```

Terminal 1 (escritor):

```
[c2211893@grajau lab6]$ gcc -Wall -o w writer.c
[c2211893@grajau lab6]$ gcc -Wall -o r reader.c
[c2211893@grajau lab6]$ ./w
```

```
Escritor: mensagem 1
Escritor: mensagem 2
Escritor: mensagem 3
Escritor: mensagem 4
Escritor: mensagem 5
Escritor: mensagem 6
Escritor: mensagem 7
Escritor: mensagem 8
Escritor: mensagem 9
Escritor: mensagem 10
Escritor: mensagem 11
Escritor: mensagem 12
Escritor: mensagem 13
Escritor: mensagem 14
Escritor: mensagem 15
Escritor: mensagem 16
Escritor: mensagem 17
Escritor: mensagem 18
Escritor: mensagem 19
Escritor: mensagem 20
Escritor: mensagem 21
Escritor: mensagem 22
Escritor: mensagem 23
Escritor: mensagem 24
Escritor: mensagem 25
Escritor: mensagem 26
Escritor: mensagem 27
Escritor: mensagem 28
Escritor: mensagem 29
Escritor: mensagem 30
Escritor: mensagem 31
Escritor: mensagem 32
Escritor: mensagem 33
Escritor: mensagem 34
Escritor: mensagem 35
Escritor: mensagem 36
Escritor: mensagem 37
Escritor: mensagem 38
Escritor: mensagem 39
Escritor: mensagem 40
Escritor: mensagem 41
Escritor: mensagem 42
Escritor: mensagem 43
Escritor: mensagem 44
Escritor: mensagem 45
Escritor: mensagem 46
Escritor: mensagem 47
Escritor: mensagem 48
Escritor: mensagem 49
Escritor: mensagem 50
```

Escreitor: mensagem 51
Escreitor: mensagem 52
Escreitor: mensagem 53
Escreitor: mensagem 54
Escreitor: mensagem 55
Escreitor: mensagem 56
Escreitor: mensagem 57
Escreitor: mensagem 58
Escreitor: mensagem 59
Escreitor: mensagem 60
Escreitor: mensagem 61
Escreitor: mensagem 62
Escreitor: mensagem 63
Escreitor: mensagem 64
Escreitor: mensagem 65
Escreitor: mensagem 66
Escreitor: mensagem 67
Escreitor: mensagem 68
Escreitor: mensagem 69
Escreitor: mensagem 70
Escreitor: mensagem 71
Escreitor: mensagem 72
Escreitor: mensagem 73
Escreitor: mensagem 74
Escreitor: mensagem 75
Escreitor: mensagem 76
Escreitor: mensagem 77
Escreitor: mensagem 78
Escreitor: mensagem 79
Escreitor: mensagem 80
Escreitor: mensagem 81
Escreitor: mensagem 82
Escreitor: mensagem 83
Escreitor: mensagem 84
Escreitor: mensagem 85
Escreitor: mensagem 86
Escreitor: mensagem 87
Escreitor: mensagem 88
Escreitor: mensagem 89
Escreitor: mensagem 90
Escreitor: mensagem 91
Escreitor: mensagem 92
Escreitor: mensagem 93
Escreitor: mensagem 94
Escreitor: mensagem 95
Escreitor: mensagem 96
Escreitor: mensagem 97
Escreitor: mensagem 98

```
Esritor: mensagem 99
Esritor: mensagem 100
Esritor: mensagem 101
Esritor: mensagem 102
Esritor: mensagem 103
Esritor: mensagem 104
Esritor: mensagem 105
Esritor: mensagem 106
Esritor: mensagem 107
Esritor: mensagem 108
Esritor: mensagem 109
Esritor: mensagem 110
Esritor: mensagem 111
Esritor: mensagem 112
Esritor: mensagem 113
Esritor: mensagem 114
Esritor: mensagem 115
Esritor: mensagem 116
Esritor: mensagem 117
Esritor: mensagem 118
Esritor: mensagem 119
Esritor: mensagem 120
Esritor: mensagem 121
Esritor: mensagem 122
Esritor: mensagem 123
Esritor: mensagem 124
Esritor: mensagem 125
Esritor: mensagem 126
Esritor: mensagem 127
Esritor: mensagem 128
[c2211893@grajau lab6]$
```

Terminal 2 (leitor):

```
[c2211893@grajau lab6]$ ./r
```

```
Leitor: mensagem 1
```

```
Leitor: mensagem 2
```

```
Leitor: mensagem 3
```

```
Leitor: mensagem 4
```

```
Leitor: mensagem 5
```

```
Leitor: mensagem 6
```

```
Leitor: mensagem 7
```

```
Leitor: mensagem 8
```

```
Leitor: mensagem 9
```

```
Leitor: mensagem 10
```

```
Leitor: mensagem 11
```

```
Leitor: mensagem 12
```

```
Leitor: mensagem 13
```

```
Leitor: mensagem 14
```

```
Leitor: mensagem 15
```

```
Leitor: mensagem 16
```

```
Leitor: mensagem 17
```

```
Leitor: mensagem 18
```

```
Leitor: mensagem 19
```

```
Leitor: mensagem 20
```

```
Leitor: mensagem 21
```

```
Leitor: mensagem 22
```

```
Leitor: mensagem 23
```

```
Leitor: mensagem 24
```

```
Leitor: mensagem 25
```

```
Leitor: mensagem 26
```

```
Leitor: mensagem 27
```

```
Leitor: mensagem 28
```

```
Leitor: mensagem 29
```

```
Leitor: mensagem 30
```

```
Leitor: mensagem 31
```

```
Leitor: mensagem 32
```

```
Leitor: mensagem 33
```

```
Leitor: mensagem 34
```

```
Leitor: mensagem 35
```

```
Leitor: mensagem 36
```

```
Leitor: mensagem 37
```

```
Leitor: mensagem 38
```

```
Leitor: mensagem 39
```

```
Leitor: mensagem 40
```

```
Leitor: mensagem 41
```

```
Leitor: mensagem 42
```

```
Leitor: mensagem 43
```

```
Leitor: mensagem 44
```

```
Leitor: mensagem 45
```

```
Leitor: mensagem 46
```

```
Leitor: mensagem 47
```

```
Leitor: mensagem 48
```

```
Leitor: mensagem 49
```

```
Leitor: mensagem 50
```

```
Leitor: mensagem 51
```

```
Leitor: mensagem 52
```

Leitor: mensagem 53
Leitor: mensagem 54
Leitor: mensagem 55
Leitor: mensagem 56
Leitor: mensagem 57
Leitor: mensagem 58
Leitor: mensagem 59
Leitor: mensagem 60
Leitor: mensagem 61
Leitor: mensagem 62
Leitor: mensagem 63
Leitor: mensagem 64
Leitor: mensagem 65
Leitor: mensagem 66
Leitor: mensagem 67
Leitor: mensagem 68
Leitor: mensagem 69
Leitor: mensagem 70
Leitor: mensagem 71
Leitor: mensagem 72
Leitor: mensagem 73
Leitor: mensagem 74
Leitor: mensagem 75
Leitor: mensagem 76
Leitor: mensagem 77
Leitor: mensagem 78
Leitor: mensagem 79
Leitor: mensagem 80
Leitor: mensagem 81
Leitor: mensagem 82
Leitor: mensagem 83
Leitor: mensagem 84
Leitor: mensagem 85
Leitor: mensagem 86
Leitor: mensagem 87
Leitor: mensagem 88
Leitor: mensagem 89
Leitor: mensagem 90
Leitor: mensagem 91
Leitor: mensagem 92
Leitor: mensagem 93
Leitor: mensagem 94
Leitor: mensagem 95
Leitor: mensagem 96
Leitor: mensagem 97
Leitor: mensagem 98
Leitor: mensagem 99
Leitor: mensagem 100


```
Leitor: mensagem 100  
Leitor: mensagem 101  
Leitor: mensagem 102  
Leitor: mensagem 103  
Leitor: mensagem 104  
Leitor: mensagem 105  
Leitor: mensagem 106  
Leitor: mensagem 107  
Leitor: mensagem 108  
Leitor: mensagem 109  
Leitor: mensagem 110  
Leitor: mensagem 111  
Leitor: mensagem 112  
Leitor: mensagem 113  
Leitor: mensagem 114  
Leitor: mensagem 115  
Leitor: mensagem 116  
Leitor: mensagem 117  
Leitor: mensagem 118  
Leitor: mensagem 119  
Leitor: mensagem 120  
Leitor: mensagem 121  
Leitor: mensagem 122  
Leitor: mensagem 123  
Leitor: mensagem 124  
Leitor: mensagem 125  
Leitor: mensagem 126  
Leitor: mensagem 127  
Leitor: mensagem 128  
[c2211893@grajau lab6]$
```

Explicação:

Os programas só tem essa saída se os dois rodam. writer.c cria e inicializa os semáforos e a memória compartilhada. Ele escreve 128 mensagens na memória compartilhada, esperando que o leitor leia cada mensagem antes de escrever a próxima. reader.c é responsável por ler as 128 mensagens da memória compartilhada. Ele aguarda a liberação do semáforo pelo processo escritor para ler cada mensagem. A sincronização entre os dois processos é garantida através do uso de dois semáforos.

b)

Shared 2:

```
// shared.h
#ifndef SHARED_H
#define SHARED_H

#include <semaphore.h>
#include <sys/types.h>
#include <sys/ipc.h>
#include <sys/shm.h>

#define MAX_MSG_LENGTH 16
#define BUFFER_SIZE 8

// Estrutura do buffer circular na memória compartilhada
typedef struct {
    char messages[BUFFER_SIZE][MAX_MSG_LENGTH];
    int in; // Índice de escrita
    int out; // Índice de leitura
} SharedBuffer;

#define SHM_KEY 1234
#define SEM_EMPTY_KEY "/sem_empty"
#define SEM_FULL_KEY "/sem_full"
#define SEM_MUTEX_KEY "/sem_mutex"

#endif // SHARED_H
```

Writer 2:

```
#include "shared.h"
#include <fcntl.h>
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <sys/mman.h>
#include <unistd.h>

int main() {
    // Criação dos semáforos
```

```

sem_t *sem_empty = sem_open(SEM_EMPTY_KEY, O_CREAT, 0666,
BUFFER_SIZE); // Controle de espaços vazios
if (sem_empty == SEM_FAILED) {
    perror("Erro ao abrir sem_empty");
    exit(1);
}
sem_t *sem_full = sem_open(SEM_FULL_KEY, O_CREAT, 0666, 0); //
Controle de mensagens presentes
if (sem_full == SEM_FAILED) {
    perror("Erro ao abrir sem_full");
    exit(1);
}
sem_t *sem_mutex = sem_open(SEM_MUTEX_KEY, O_CREAT, 0666, 1); //
Exclusão mútua
if (sem_mutex == SEM_FAILED) {
    perror("Erro ao abrir sem_mutex");
    exit(1);
}
// Criação da memória compartilhada
int shm_id = shmget(5789, sizeof(SharedBuffer), IPC_CREAT | 0666);
if (shm_id == -1) {
    perror("Erro ao criar memória compartilhada");
    exit(1);
}
SharedBuffer *buffer = (SharedBuffer *)shmat(shm_id, NULL, 0);

buffer->in = 0;

for (int i = 1; i <= 128; i++) {
    // Gera a mensagem
    char msg[MAX_MSG_LENGTH];
    snprintf(msg, MAX_MSG_LENGTH, "mensagem %d", i);

    // Espera até haver espaço no buffer
    sem_wait(sem_empty);
    sem_wait(sem_mutex);

    // Insere a mensagem no buffer

```

```

strcpy(buffer->messages[buffer->in], msg);
printf("Escritor: Enviou %s\n", msg);
buffer->in = (buffer->in + 1) % BUFFER_SIZE;

sem_post(sem_mutex);
sem_post(sem_full);

sleep(1); // Simula atraso para visualizar o funcionamento
}

// Limpeza
sem_close(sem_empty);
sem_close(sem_full);
sem_close(sem_mutex);
shmdt(buffer);
sem_unlink(SEM_EMPTY_KEY);
sem_unlink(SEM_FULL_KEY);
sem_unlink(SEM_MUTEX_KEY);
return 0;
}

```

Reader 2:

```

#include "shared.h"
#include <fcntl.h>
#include <stdio.h>
#include <stdlib.h>
#include <sys/mman.h>
#include <unistd.h>

int main() {
    // Abertura dos semáforos
    sem_t *sem_empty = sem_open(SEM_EMPTY_KEY, 0);
    if (sem_empty == SEM_FAILED) {
        perror("Erro ao abrir sem_empty");
        exit(1);
    }
}

```

```

}
sem_t *sem_full = sem_open(SEM_FULL_KEY, 0);
if (sem_full == SEM_FAILED) {
    perror("Erro ao abrir sem_full");
    exit(1);
}
sem_t *sem_mutex = sem_open(SEM_MUTEX_KEY, 0);
if (sem_mutex == SEM_FAILED) {
    perror("Erro ao abrir sem_mutex");
    exit(1);
}

// Acesso à memória compartilhada
int shm_id = shmget(5789, sizeof(SharedBuffer), 0666);
SharedBuffer *buffer = (SharedBuffer *)shmat(shm_id, NULL, 0);

buffer->out = 0;

for (int i = 1; i <= 128; i++) {
    // Espera até haver uma mensagem no buffer
    sem_wait(sem_full);
    sem_wait(sem_mutex);

    // Lê a mensagem do buffer
    printf("Leitor: Recebeu %s\n", buffer->messages[buffer->out]);
    buffer->out = (buffer->out + 1) % BUFFER_SIZE;

    sem_post(sem_mutex);
    sem_post(sem_empty);

    sleep(2); // Simula atraso para visualizar o funcionamento
}

// Limpeza
sem_close(sem_empty);
sem_close(sem_full);
sem_close(sem_mutex);
shmdt(buffer);

```

```
return 0;  
}
```

Saídas:

```
~/testebusca$ gcc -Wall -o w writer2.c
~/testebusca$ gcc -Wall -o r reader2.c
~/testebusca$ ./w
Escritor: Enviou mensagem 1
Escritor: Enviou mensagem 2
Escritor: Enviou mensagem 3
Escritor: Enviou mensagem 4
Escritor: Enviou mensagem 5
Escritor: Enviou mensagem 6
Escritor: Enviou mensagem 7
Escritor: Enviou mensagem 8
Escritor: Enviou mensagem 9
Escritor: Enviou mensagem 10
Escritor: Enviou mensagem 11
Escritor: Enviou mensagem 12
Escritor: Enviou mensagem 13
Escritor: Enviou mensagem 14
Escritor: Enviou mensagem 15
Escritor: Enviou mensagem 16
Escritor: Enviou mensagem 17
Escritor: Enviou mensagem 18
Escritor: Enviou mensagem 19
Escritor: Enviou mensagem 20
Escritor: Enviou mensagem 21
Escritor: Enviou mensagem 22
Escritor: Enviou mensagem 23
Escritor: Enviou mensagem 24
Escritor: Enviou mensagem 25
Escritor: Enviou mensagem 26
Escritor: Enviou mensagem 27
Escritor: Enviou mensagem 28
Escritor: Enviou mensagem 29
Escritor: Enviou mensagem 30
Escritor: Enviou mensagem 31
Escritor: Enviou mensagem 32
Escritor: Enviou mensagem 33
Escritor: Enviou mensagem 34
Escritor: Enviou mensagem 35
Escritor: Enviou mensagem 36
Escritor: Enviou mensagem 37
Escritor: Enviou mensagem 38
Escritor: Enviou mensagem 39
Escritor: Enviou mensagem 40
Escritor: Enviou mensagem 41
Escritor: Enviou mensagem 42
Escritor: Enviou mensagem 43
Escritor: Enviou mensagem 44
Escritor: Enviou mensagem 45
```

Escrivor: Enviou mensagem 46
Escrivor: Enviou mensagem 47
Escrivor: Enviou mensagem 48
Escrivor: Enviou mensagem 49
Escrivor: Enviou mensagem 50
Escrivor: Enviou mensagem 51
Escrivor: Enviou mensagem 52
Escrivor: Enviou mensagem 53
Escrivor: Enviou mensagem 54
Escrivor: Enviou mensagem 55
Escrivor: Enviou mensagem 56
Escrivor: Enviou mensagem 57
Escrivor: Enviou mensagem 58
Escrivor: Enviou mensagem 59
Escrivor: Enviou mensagem 60
Escrivor: Enviou mensagem 61
Escrivor: Enviou mensagem 62
Escrivor: Enviou mensagem 63
Escrivor: Enviou mensagem 64
Escrivor: Enviou mensagem 65
Escrivor: Enviou mensagem 66
Escrivor: Enviou mensagem 67
Escrivor: Enviou mensagem 68
Escrivor: Enviou mensagem 69
Escrivor: Enviou mensagem 70
Escrivor: Enviou mensagem 71
Escrivor: Enviou mensagem 72
Escrivor: Enviou mensagem 73
Escrivor: Enviou mensagem 74
Escrivor: Enviou mensagem 75
Escrivor: Enviou mensagem 76
Escrivor: Enviou mensagem 77
Escrivor: Enviou mensagem 78
Escrivor: Enviou mensagem 79
Escrivor: Enviou mensagem 80
Escrivor: Enviou mensagem 81
Escrivor: Enviou mensagem 82
Escrivor: Enviou mensagem 83
Escrivor: Enviou mensagem 84
Escrivor: Enviou mensagem 85
Escrivor: Enviou mensagem 86
Escrivor: Enviou mensagem 87
Escrivor: Enviou mensagem 88
Escrivor: Enviou mensagem 89
Escrivor: Enviou mensagem 90
Escrivor: Enviou mensagem 91
Escrivor: Enviou mensagem 92
Escrivor: Enviou mensagem 93
Escrivor: Enviou mensagem 94
Escrivor: Enviou mensagem 95


```
Esritor: Enviou mensagem 96
Esritor: Enviou mensagem 97
Esritor: Enviou mensagem 98
Esritor: Enviou mensagem 99
Esritor: Enviou mensagem 100
Esritor: Enviou mensagem 101
Esritor: Enviou mensagem 102
Esritor: Enviou mensagem 103
Esritor: Enviou mensagem 104
Esritor: Enviou mensagem 105
Esritor: Enviou mensagem 106
Esritor: Enviou mensagem 107
Esritor: Enviou mensagem 108
Esritor: Enviou mensagem 109
Esritor: Enviou mensagem 110
Esritor: Enviou mensagem 111
Esritor: Enviou mensagem 112
Esritor: Enviou mensagem 113
Esritor: Enviou mensagem 114
Esritor: Enviou mensagem 115
Esritor: Enviou mensagem 116
Esritor: Enviou mensagem 117
Esritor: Enviou mensagem 118
Esritor: Enviou mensagem 119
Esritor: Enviou mensagem 120
Esritor: Enviou mensagem 121
Esritor: Enviou mensagem 122
Esritor: Enviou mensagem 123
Esritor: Enviou mensagem 124
Esritor: Enviou mensagem 125
Esritor: Enviou mensagem 126
Esritor: Enviou mensagem 127
Esritor: Enviou mensagem 128
```

Terminal 2:

```
[c2310289@volga lab6S0]$ ./r
```

```
Leitor: Recebeu mensagem 1
```

```
Leitor: Recebeu mensagem 2
```

```
Leitor: Recebeu mensagem 3
```

```
Leitor: Recebeu mensagem 4
```

```
Leitor: Recebeu mensagem 5
```

```
Leitor: Recebeu mensagem 6
```

```
Leitor: Recebeu mensagem 7
```

```
Leitor: Recebeu mensagem 8
```

```
Leitor: Recebeu mensagem 9
```

```
Leitor: Recebeu mensagem 10
```

```
Leitor: Recebeu mensagem 11
```

```
Leitor: Recebeu mensagem 12
```

```
Leitor: Recebeu mensagem 13
```

```
Leitor: Recebeu mensagem 14
```

```
Leitor: Recebeu mensagem 15
```

```
Leitor: Recebeu mensagem 16
```

```
Leitor: Recebeu mensagem 17
```

```
Leitor: Recebeu mensagem 18
```

```
Leitor: Recebeu mensagem 19
```

```
Leitor: Recebeu mensagem 20
```

```
Leitor: Recebeu mensagem 21
```

```
Leitor: Recebeu mensagem 22
```

```
Leitor: Recebeu mensagem 23
```

```
Leitor: Recebeu mensagem 24
```

```
Leitor: Recebeu mensagem 25
```

```
Leitor: Recebeu mensagem 26
```

```
Leitor: Recebeu mensagem 27
```

```
Leitor: Recebeu mensagem 28
```

```
Leitor: Recebeu mensagem 29
```

```
Leitor: Recebeu mensagem 30
```

```
Leitor: Recebeu mensagem 31
```

```
Leitor: Recebeu mensagem 32
```

```
Leitor: Recebeu mensagem 33
```

```
Leitor: Recebeu mensagem 34
```

```
Leitor: Recebeu mensagem 35
```

```
Leitor: Recebeu mensagem 36
```

```
Leitor: Recebeu mensagem 37
```

```
Leitor: Recebeu mensagem 38
```

```
Leitor: Recebeu mensagem 39
```

```
Leitor: Recebeu mensagem 40
```

```
Leitor: Recebeu mensagem 41
```

```
Leitor: Recebeu mensagem 42
```

```
Leitor: Recebeu mensagem 43
```

```
Leitor: Recebeu mensagem 44
```

```
Leitor: Recebeu mensagem 45
```

```
Leitor: Recebeu mensagem 46
```

```
Leitor: Recebeu mensagem 47
```

```
Leitor: Recebeu mensagem 48
```

```
Leitor: Recebeu mensagem 49
```

```
Leitor: Recebeu mensagem 50
```

```
Leitor: Recebeu mensagem 51
```

```
Leitor: Recebeu mensagem 52
```

Leitor: Recebeu mensagem 53
Leitor: Recebeu mensagem 54
Leitor: Recebeu mensagem 55
Leitor: Recebeu mensagem 56
Leitor: Recebeu mensagem 57
Leitor: Recebeu mensagem 58
Leitor: Recebeu mensagem 59
Leitor: Recebeu mensagem 60
Leitor: Recebeu mensagem 61
Leitor: Recebeu mensagem 62
Leitor: Recebeu mensagem 63
Leitor: Recebeu mensagem 64
Leitor: Recebeu mensagem 65
Leitor: Recebeu mensagem 66
Leitor: Recebeu mensagem 67
Leitor: Recebeu mensagem 68
Leitor: Recebeu mensagem 69
Leitor: Recebeu mensagem 70
Leitor: Recebeu mensagem 71
Leitor: Recebeu mensagem 72
Leitor: Recebeu mensagem 73
Leitor: Recebeu mensagem 74
Leitor: Recebeu mensagem 75
Leitor: Recebeu mensagem 76
Leitor: Recebeu mensagem 77
Leitor: Recebeu mensagem 78
Leitor: Recebeu mensagem 79
Leitor: Recebeu mensagem 80
Leitor: Recebeu mensagem 81
Leitor: Recebeu mensagem 82
Leitor: Recebeu mensagem 83
Leitor: Recebeu mensagem 84
Leitor: Recebeu mensagem 85
Leitor: Recebeu mensagem 86
Leitor: Recebeu mensagem 87
Leitor: Recebeu mensagem 88
Leitor: Recebeu mensagem 89
Leitor: Recebeu mensagem 90
Leitor: Recebeu mensagem 91
Leitor: Recebeu mensagem 92
Leitor: Recebeu mensagem 93
Leitor: Recebeu mensagem 94
Leitor: Recebeu mensagem 95
Leitor: Recebeu mensagem 96
Leitor: Recebeu mensagem 97
Leitor: Recebeu mensagem 98
Leitor: Recebeu mensagem 99
Leitor: Recebeu mensagem 100
Leitor: Recebeu mensagem 101
Leitor: Recebeu mensagem 102
Leitor: Recebeu mensagem 103
Leitor: Recebeu mensagem 104
Leitor: Recebeu mensagem 105

```
Leitor: Recebeu mensagem 106
Leitor: Recebeu mensagem 107
Leitor: Recebeu mensagem 108
Leitor: Recebeu mensagem 109
Leitor: Recebeu mensagem 110
Leitor: Recebeu mensagem 111
Leitor: Recebeu mensagem 112
Leitor: Recebeu mensagem 113
Leitor: Recebeu mensagem 114
Leitor: Recebeu mensagem 115
Leitor: Recebeu mensagem 116
Leitor: Recebeu mensagem 117
Leitor: Recebeu mensagem 118
Leitor: Recebeu mensagem 119
Leitor: Recebeu mensagem 120
Leitor: Recebeu mensagem 121
Leitor: Recebeu mensagem 122
Leitor: Recebeu mensagem 123
Leitor: Recebeu mensagem 124
Leitor: Recebeu mensagem 125
Leitor: Recebeu mensagem 126
Leitor: Recebeu mensagem 127
Leitor: Recebeu mensagem 128
[c2310289@volga lab6S0]$
```

Explicação:

Este programa implementa a troca de mensagens entre dois processos assíncronos: um escritor e um leitor. A comunicação ocorre através de um buffer circular, implementado na memória compartilhada. Os processos usam semáforos para gerenciar o acesso ao buffer, garantindo sincronização correta, mesmo quando ambos os processos tentam acessar o buffer ao mesmo tempo. O buffer pode armazenar até 8 mensagens de forma concorrente. O código utiliza três semáforos: `sem_empty` (controla o número de espaços vazios no buffer), `sem_full` (controla o número de mensagens presentes no buffer) e `sem_mutex` (controla a exclusão mútua, garantindo que apenas um processo (escritor ou leitor) acesse o buffer por vez). O buffer circular é essencial para garantir que múltiplas mensagens possam ser escritas no buffer pelo processo escritor e lidas pelo processo leitor, mesmo que um processo seja mais rápido do que o outro. Quando o escritor atinge o final do buffer, ele retorna ao início (índice 0) para sobrescrever os dados já lidos pelo leitor, permitindo uma utilização eficiente do espaço. A exclusão mútua no acesso ao buffer é mantida pelo semáforo `sem_mutex`, para evitar condições de corrida.

Diferença entre os dois:

Do primeiro jeito é quase instantânea a saída, porém desse segundo jeito, os programas demoram mais.