Int followers_waiting=0; // keep count of waiting dancers

Int leaders_waiting=0; // keep count of waiting dancers

Mutex leader_counter_mutex, follower_counter_mutex; //to keep counters thread safe

Semaphore stage_semaphores[4]; //used for controlling the entry and the exit to the stage

// first two semaphores are to check the entry to the stage and the last two are to check the exit of the stage. They all start locked

Leader()

{

```
lock(leader_counter_mutex);
leaders_waiting++;
pthread_mutex_unlock(&leader_counter_mutex);

signal(stage_semaphores[0]);

lock(follower_counter_mutex);
if (followers_waiting) print(followers left message);
else print(leaders waiting message);

unlock(follower_counter_mutex);

wait(stage_semaphores[1]);
leaders_waiting--;
print(dancing together message);
signal(stage_semaphore[2]);
wait(stage_semaphore[3]);

print(leave stage message);
```

}

Follower()

{

```
lock(follower_counter_mutex);
followers_waiting++;
pthread_mutex_unlock(&leader_counter_mutex);

signal(stage_semaphores[1]);

lock(leader_counter_mutex);
if (leaders_waiting) print(leaders left message);
else print(followers waiting message);

unlock(leader_counter_mutex);

wait(stage_semaphores[0]);
followers_waiting--;
print(dancing together message);
signal(stage_semaphore[3]);
wait(stage_semaphore[2]);

print(leave stage message);
```

}