

Concurrency – Deadlock

Alexandre David

1. Exercises

1. Exercise 6.1.
2. Exercise 6.2.
3. Exercise 6.4.

2. Problem: Rush Hour Game

The game consists of a small board of 6x6 positions where cars of length 2 or 3 are put horizontally or vertically. The goal is to get the red car exit. Cars may move forward or backward (but not turn or go on the side) if there is free space. Model the game with either an ERROR or a deadlock state being the goal. You can download *car-exo.lts* from the web page to start with or you can do it as you wish. The example has 2 instances of the game, the first is too hard for LTSA and the second has a solution that you can find 1. The argument to the process is the position (of the front or back) of the car. The instances are coded as follows: $c[id].h[row]:CAR2(col)$ or $c[id].v[col]:CAR2(row)$ (for a car of length 2) where id is the ID of the car, v is for a vertical car, and h is for an horizontal car.

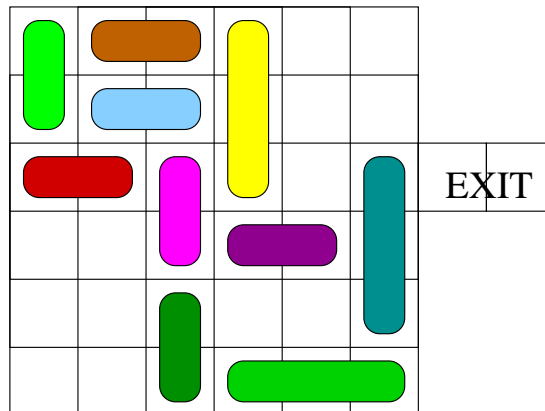


Figure 1: Instance for which you can find a solution.