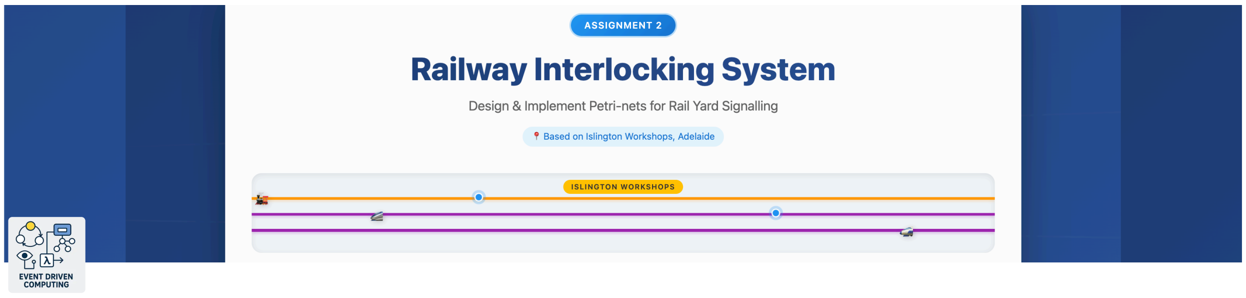

ASSIGNMENT 2
Railway Interlocking System
Design & Implement Petri-nets for Rail Yard Signalling
Based on Islington Workshops, Adelaide
ISLINGTON WORKSHOPS
EVENT DRIVEN COMPUTING

Programming Assignment 2 Conflict Scenario and feedback

Hello Students,

I just wanted to add a clarification note about autograder's feedback in Programming Assignment 2:

1. Programming Assignment overview:

In this assignment, we have to simulate a snippet of a train network acting as a concurrent system, in which each train is treated as an independent subsystem. Now, the subsystems(train) are independent as they are a self-contained set of processes(behaviour: to move or stay from a section) and the resources(track section and intersections) they use can execute(in our case, the transition is possible) or deadlock(further explained below) without any influence/information from the rest of the subsystems(information-hiding).

Check the Programming Assignment 2 for full Problem Statement.

2. Conflict scenarios

In this system, as each train is an independent subsystem, it can lead to some interesting scenarios:

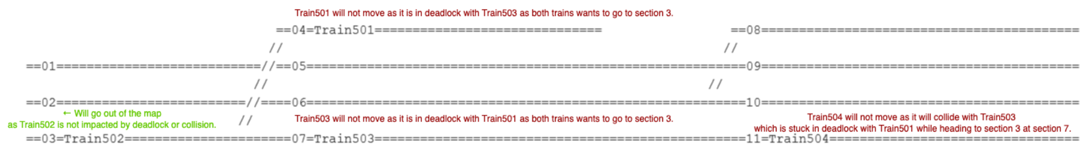
- System-wide deadlock
- Local deadlock
- Collision

2.1. System-wide deadlock

- Labelled as “deadlock”.
- This happens when every train in the network is unable to execute(here, unable to move to the desired train section when asked to move due to below scenarios).
- This is what can be referred to as an entire network being in “deadlock” (FSA Notes, pg. 75)

2.2. Local deadlock

- Also labelled as “deadlock” in autograder.
- This happens when a set/group of trains are each waiting on a conflict-free resource(for the assignment, an empty track section or track section which will become empty in the same movement cycle at the intersection).
- With this we have 2 possible scenarios that can happen:
    - When two train at an intersection wants to access same track section but can not move as they wait on each other; This causes a stalemate where no progress can be made by either of the train(subsystems), i.e., none of them can move until the other moves first(which can be done by the network’s intervention when it intentionally moves either of the trains and not the others).
        - For example the below scenario,



Train501 will not move as it is in deadlock with Train503 as both trains wants to go to section 3.

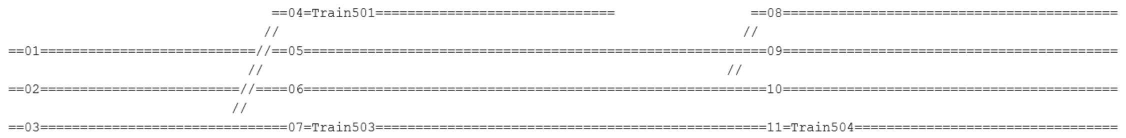
```
                                 ==04=Train501==============================            ==08==========================================
                                //                                                     //
==01==========================//==05=====================================================09==========================================
                              //                                                     //
==02========================//====06=====================================================10==========================================
                           //
==03=Train502======================07=Train503==============================================11=Train504================================
```

← Will go out of the map
as Train502 is not impacted by deadlock or collision.

Train503 will not move as it is in deadlock with Train501 as both trains wants to go to section 3.

Train504 will not move as it will collide with Train503
which is stuck in deadlock with Train501 while heading to section 3 at section 7.

Expected Result when moving Train501, Train502, Train503, Train504:



```
                                 ==04=Train501==============================            ==08==========================================
                                //                                                     //
==01==========================//==05=====================================================09==========================================
                              //                                                     //
==02========================//====06=====================================================10==========================================
                           //
==03===============================07=Train503==============================================11=Train504================================
```

- Where 2 train are right next to each other and want to go to each-others track section but can not move otherwise they will collide(labeled as "collision" see below for more details).

2.3. Collision

- Labelled as “collision”. This is a special case of local deadlock, where two trains attempted to occupy the other’s section in the same movement cycle, or a train attempts to move to a section where a train is in a state of local deadlock or is staying at a section as they were not asked to move.

Note: Local deadlock or collision doesn’t mean the entire network is in a deadlock, as the network might still have a train(subsystem) that can execute(trains which are asked to move and can move and should move). However, if the local deadlock is not resolved and not broken, they will be stuck in their sections and eventually other trains may reach those sections and also become blocked(cause a collision if moved), which cascades into a system-wide deadlock where no train can move. So, as there is a group/set of trains which are in a stalemate, the entire system will be in a state of system-wide deadlock; if the network asks to move all of the trains all the time.

3. Conflict scenario resolution

This can be resolved by

- Prioritisation of trains
- Intentional Transition firing:

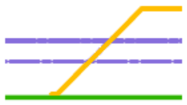
3.1. Prioritisation of trains

- The simulation does prioritise passenger trains and their transitions over freight trains’ transitions.
  - Do note that the assignment does not specify any prioritisation rule for breaking ties among the same train type(2 trains of the same type, i.e., either passenger-passenger or freight-freight are treated equally). In our network, we do not have any track section which can be occupied by both freight trains and passenger trains, either by entry, transition or intersection.
  - However, we do have a special scenario (also refer to the attached snippet from the assignment problem statement) in the freight train line where it has a two-way intersection at sections 3, 4 and 7 and the track between sections 3 and 4 crosses 2 passenger lines(1→5 and 2<-6).
    - Labelled as “block”.
    - Do note that at this intersection, the passenger train should always be given priority over the freight train.
    - So, if there is a freight train going from 3→4 or 3<-4, it can only move if there is no train going 1→5 and 2<-6 in the same movement cycle.

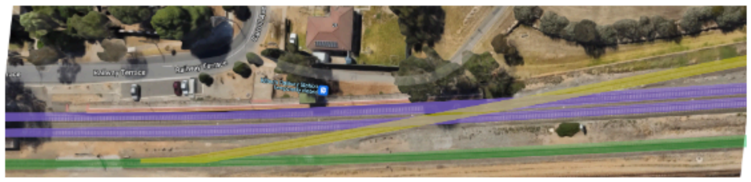
Junctions

There are 2 main junctions in the corridor.

One where the freight line splits and crosses the passenger line:



Passenger trains should always have priority at this junction.



3.2. Intentional transition firing

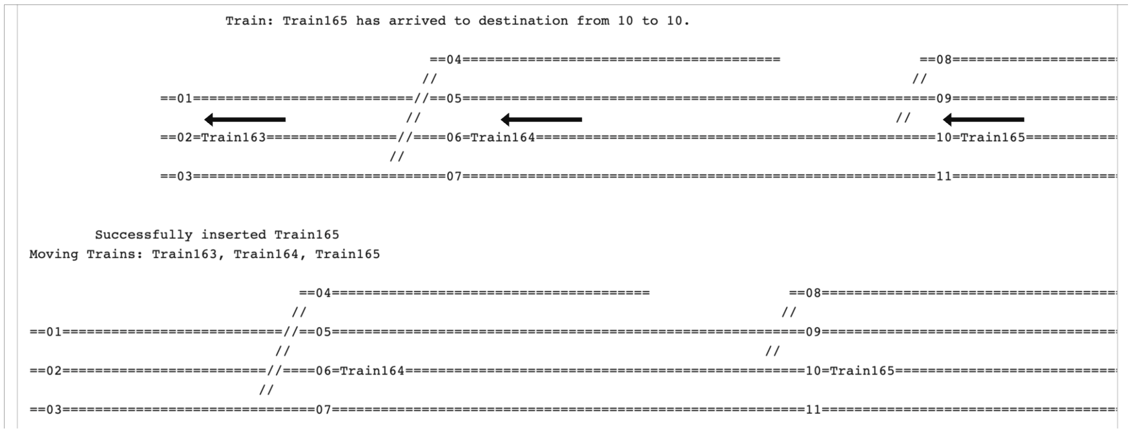
- If the network intentionally tells the train to move in a manner that would resolve the deadlock. Now, your program allows the autograder to call which trains to move in a movement cycle, which then your program attempts to move and should move if possible.

4. Autograder testing

- The autograder tests various scenarios by moving the trains in position and then move all the trains. Some of these trains may move while others may not due to conflict. The autograder will test any combination of status successful movement, deadlock(local or network), collision and block.
- If any test scenario fails, the autograder will inform you in red test cases. Do note that there are some test cases which are hidden, and you are expected to test your code throughly and have diverse set of test case scenarios to find them.
- For any visible test case failure, do check the logs to find out what trains are being inserted and how the map is set and then when the map is set, move all the trains to test your scenarios.
    - There can be 3 main cases of the failure(there might be more):
        - Failure to set the map properly, this failure will cause multiple test cases to fail.
            - This can be found by unexpected termination of testing log

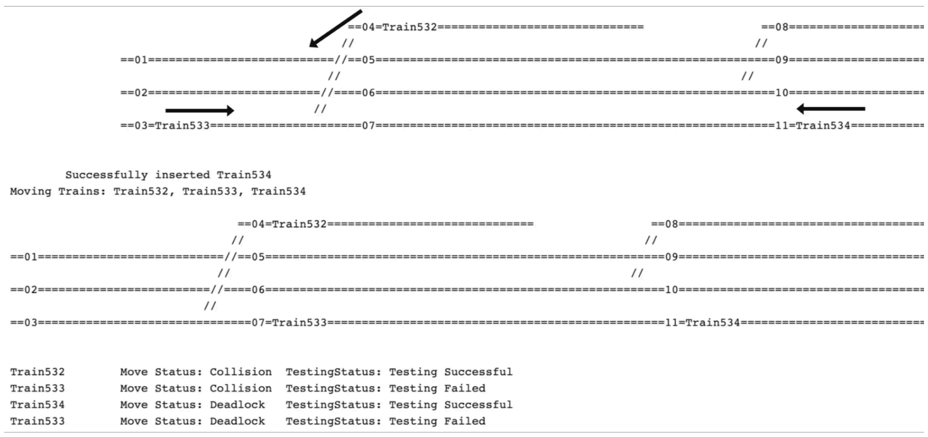
```
Test failed.

Detailed result for Section01InterlockingImplTestingFinal: testcase024
Result: Test Failed
Log file content:
Adding Trains in Bulk:
        Inserting Train532
                Train: Train532 will go from 4 to 3 but will stop at 4 temporarily.
                        Train: Train532 is scheduled to go from 4 to 3.

                                                 ==04=Train532==============================                 ==08===================
                                                //                                                          //
                ==01=========================//==05=========================================================09=====================
                                              //                                                          //
                ==02=========================//====06=======================================================10=====================
                                            //
                ==03=============================07=======================================================11=====================


                        Train: Train532 has arrived to destination from 4 to 4.

                                                 ==04=Train532==============================                 ==08===================
                                                //                                                          //
                ==01=========================//==05=========================================================09=====================
                                              //                                                          //
                ==02=========================//====06=======================================================10=====================
                                            //
                ==03=============================07=======================================================11=====================


        Successfully inserted Train532
        Inserting Train533
                Train: Train533 will go from 3 to 11 but will stop at 3 temporarily.
```

- Missing movement of the trains when they are asked and expected to move.
  - In the below scenario we expect Train163, Train164 and Train165 to move out through section 2.
  - Expected:
    - Train 163: Out of map,
    - Train 164: Section 2
    - Train 165: Section 6
  - Actual:
    - Train 163: Out of map
    - Train 164: Section 6 (Test Failed)
    - Train 165: Section 10 (Test Failed)



```
                        Train: Train165 has arrived to destination from 10 to 10.

                                                 ==04=======================================                 ==08====================
                                                //                                                          //
                ==01=========================//==05=========================================================09=====================
                                              //                                                          //
                ==02=Train163================//====06=Train164==============================================10=Train165============
                                            //
                ==03=============================07=======================================================11=====================


        Successfully inserted Train165
Moving Trains: Train163, Train164, Train165

                                 ==04=======================================                 ==08====================================
                                //                                                          //
==01=========================//==05=========================================================09=====================================
                              //                                                          //
==02=========================//====06=Train164==============================================10=Train165============================
                            //
==03=============================07=======================================================11=====================================
```

- Movement of trains when they are not expected to move.
  - In this case you would see Train ID, expected status(”Block”, “Collision”, “Deadlock”(local or network)), and if they move, the testing will labeled as “Testing Failed”.
  - In below example,



```
                                                 ==04=Train532==============================            ==08====================
                                                //                                                     //
                ==01===========================//==05==========================================================09====================
                                              //                                                     //
                ==02=========================//====06==========================================================10====================
                                            //
                ==03=Train533======================07==========================================================11=Train534===========


        Successfully inserted Train534
Moving Trains: Train532, Train533, Train534

                                 ==04=Train532==============================              ==08====================================
                                //                                                       //
==01===========================//==05==========================================================09====================================
                              //                                                       //
==02=========================//====06==========================================================10====================================
                            //
==03===============================07=Train533=================================================11=Train534===========================


Train532        Move Status: Collision  TestingStatus: Testing Successful
Train533        Move Status: Collision  TestingStatus: Testing Failed
Train534        Move Status: Deadlock   TestingStatus: Testing Successful
Train533        Move Status: Deadlock   TestingStatus: Testing Failed
```

- Expected Train 533 to not move as it is in local deadlock with Train 532 and collides with Train 534, but it moved.

Note: Above test cases are an example and your actual output/log might differ; So, have a look and understand your failed test logs and use these example as guide.

I hope this helps clarify the distinction between system-wide deadlock and local deadlock & collision affecting a group/subset of trains.