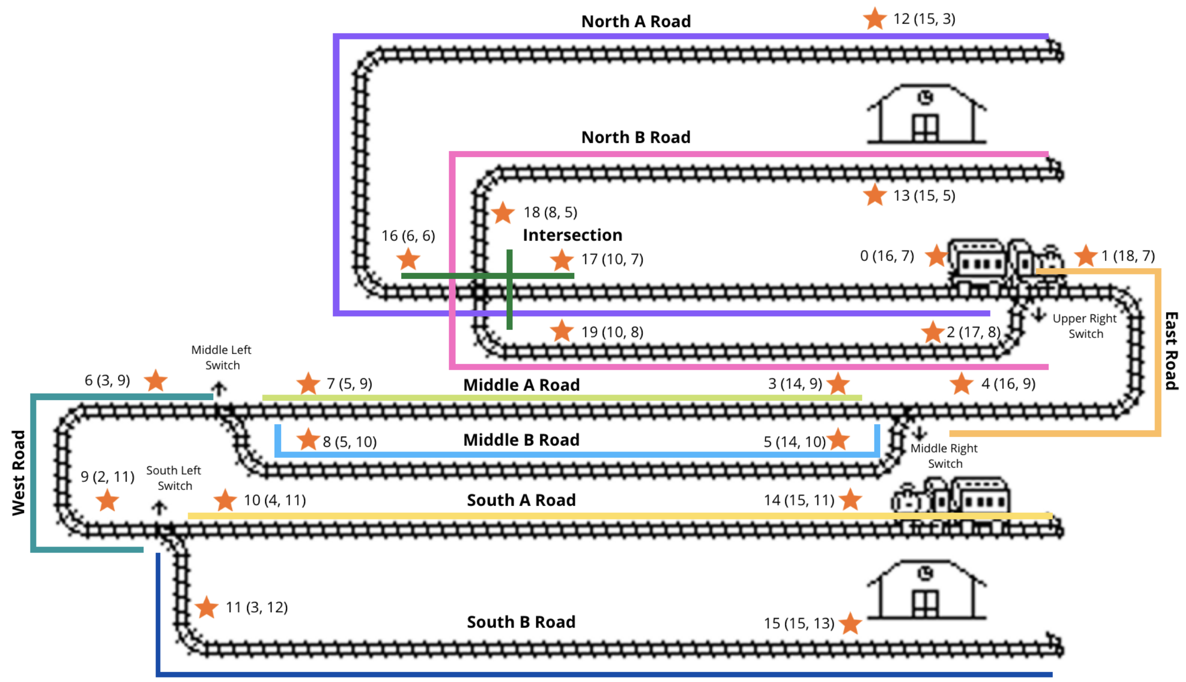
Principles of Concurrent Programming
Ricardo Escobar Gouyonnet
Doménica Rentería Berrospe
Team 44

Trainspotting

The following text will address how we managed to solve Lab 1 - Trainspotting by implementing semaphores. We modeled our solution as the following map shows:



Choice of critical sections

This solution has 9 critical sections, therefore 9 semaphores. These sections can not be shared, therefore we implemented these semaphores in order to keep them from being shared.

1. **North A Road -** On the map it's signaled by the purple color. It starts/ends on the sensor 12 and/or 0.
2. **North B Road -** On the map it's signaled by the pink color. It starts/ends on the sensor 13 and/or 2.
3. **Intersection-** On the map it's signaled by the dark green color. It starts/ends on the sensor 16-19.
4. **East Road -** On the map it's signaled by the orange color. It starts/ends on the sensor 1 and/or 4.
5. **Middle A Road -** On the map it's signaled by the light green color. It starts/ends on the sensor 3 and/or 7.
6. **Middle B Road -** On the map it's signaled by the light blue color. It starts/ends on the sensor 5 and/or 8.

7. **West Road -** On the map it's signaled by the aqua color. It starts/ends on the sensor 6 and/or 9.
8. **South A Road -** On the map it's signaled by the yellow color. It starts/ends on the sensor 10 and/or 14.
9. **South B Road -** On the map it's signaled by the dark blue color. It starts/ends on the sensor 11 and/or 15.

Placement of the sensors

This solution has **19** sensors. We placed each sensor at the beginning and/or end of each critical section. This placement was made to keep track of the position of the trains. If any of the sensors were activated, the program evaluates the situation and can make a choice depending on the scenario: activate or deactivate a semaphore, change the direction of the switches, make a stop/wait, or change the direction of the trains.

Sensors **0,** and **2** are in charge of deactivating the semaphores for **North A Road** and **North B Road;** they also evaluate if the trains should wait to avoid a crash. The logic for these sensors works as the following: if a critical section is taken by the other train, it will wait until it can acquire the semaphore setting the speed of the train to 0. If it couldn't acquire the semaphore on the first attempt, it means that a train is just going by. In order to know this, we implemented a flag. Once the train can acquire the semaphore and until it passes the sensor, the speed will be equal to 1, in order to give enough time and space for the other train to pass through the switch section and avoid an early change of the switch. It then releases the critical section making it available for the next train and changes the switch to make sure the train goes to the correct trail.

Sensor **1** is in charge of activating the semaphores for **North A Road** and **North B Road.**

Sensors **0, 1,** and **2** are in charge of changing the direction of the **Upper Right** switch.

Sensors **0, 2, 3,** and **5** are in charge of activating the **East Road** semaphore.

Sensors **1,** and **4** are in charge of deactivating the **East Road** semaphore.

Sensors **3, 4,** and **5** are in charge of changing the direction of the **Middle Right** switch.

Sensors **3** and **5** evaluate if the trains should wait to avoid crashing in the **Middle Right** switches. The logic for these sensors works as the following: if a critical section is taken by the other train, it will wait until it can acquire the semaphore setting the speed of the train to 0. If it couldn't acquire the semaphore on the first attempt, it means that a train is just going by. In order to know this, we implemented a flag. Once the train can acquire the semaphore and until it passes the sensor, the speed will be equal to 1, in order to give enough time and space for the other train to pass through the switch section and avoid an early change of the switch. It then releases the critical section making it available for the next train and changes the switch to make sure the train goes to the correct trail.

Sensors **4,** and **6** are in charge of activating the semaphores for **Middle A Road** and **Middle B Road.**

Sensors **3, 5, 7,** and **8** are in charge of deactivating the semaphores for **Middle A Road** and **Middle B Road**; they also evaluate if the trains should wait to avoid a crash.

Sensors **6, 7,** and **8** are in charge of changing the direction of the **Middle Left** switch.

Sensors **7** and **8** evaluate if the trains should wait to avoid crashing in the **Middle Left** switches. The logic for these sensors works as the following: if a critical section is taken by the other train, it will wait until it can acquire the semaphore setting the speed of the train to 0. If it couldn't acquire the semaphore on the first attempt, it means that a train is just going by. In order to know this, we implemented a flag. Once the train can acquire the semaphore and until it passes the sensor, the speed will be equal to 1, in order to give enough time and space for the other train to pass through the switch section and avoid an early change of the switch. It then releases the critical section making it available for the next train and changes the switch to make sure the train goes to the correct trail.

Sensors **7, 8, 10,** and **11** are in charge of activating the **West Road** semaphore.

Sensors **6,** and **9** are in charge of deactivating the **West Road** semaphore.

Sensors **9, 10,** and **11** are in charge of changing the direction of the **South Left** switch.

Sensor **9** is in charge of activating the semaphores for **South A Road** and **South B Road.**

Sensors **10,** and **11** are in charge of deactivating the semaphores for **South A Road** and **South B Road;** they also evaluate if the trains should wait to avoid a crash. The logic for these sensors works as the following: if a critical section is taken by the other train, it will wait until it can acquire the semaphore setting the speed of the train to 0. If it couldn't acquire the semaphore on the first attempt, it means that a train is just going by. In order to know this, we implemented a flag. Once the train can acquire the semaphore and until it passes the sensor, the speed will be equal to 1, in order to give enough time and space for the other train to pass through the switch section and avoid an early change of the switch. It then releases the critical section making it available for the next train and changes the switch to make sure the train goes to the correct trail.

Sensors **16, 17, 18,** and **19** are in charge of activating and deactivating the **Intersection** semaphore; they also evaluate if the trains should wait to avoid a crash.

Sensors **12, 13, 14,** and **15** are in charge of stopping the train for 2 seconds and changing the trains' way.

Maximum train speed and the reason for it

The maximum speed for the trains is **15**. The reason behind this is that with this speed and the sensor placement, it has enough time for the trains to reduce their speed in time in order to make a full stop and avoid derailment, and/or wait for the release of semaphores. If the trains go over this speed, in some scenarios, they can't stop in time and they derail, or crash.

Principles of Concurrent Programming
Ricardo Escobar Gouyonnet
Doménica Rentería Berrospe
Team 44

How you tested your solution

The testing was based on setting different speed values to the trains, seeing their behavior and evaluating if it works. We left the simulator working with these speeds for several minutes in order to evaluate and define the highest speed it can handle. It works for all speeds below and including our maximum speed of 15.