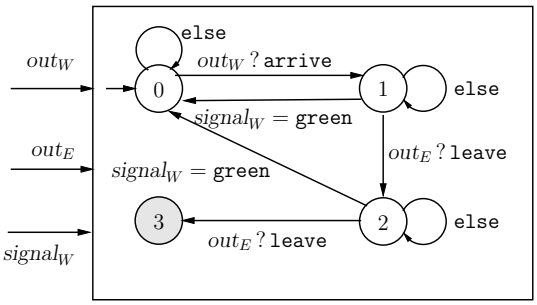
Lab 1 – NuSMV and safety properties

Exercise 1. Consider the railroad example of last lesson. You can find the SMV code of the first solution, where trains can be simultaneously on the brige, in the file `railroad_wrong.smv`.

(a) Download the file from Moodle and verify with `NuSMV` that the invariant is falsified.
(b) Modify the code of `controller` to "fix the bug". Check that the new controller satisfies the invariant.

Exercise 2. Consider the fairness monitor *WestFairMonitor* for the west train in the figure. Verify with `MuSMV` if the second attempt to design the railroad controller satisfy the fairness requirement captured by the monitor.



Exercise 3. Consider the wolf, goat and cabbage problem:

> Once upon a time a farmer went to a market and purchased a wolf, a goat, and a cabbage. On his way home, the farmer came to the bank of a river and rented a boat. But crossing the river by boat, the farmer could carry only himself and a single one of his purchases: the wolf, the goat, or the cabbage.
>
> If left unattended together, the wolf would eat the goat, or the goat would eat the cabbage.
>
> The farmer's challenge was to carry himself and his purchases to the far bank of the river, leaving each purchase intact. How did he do it?[1]

Model the wolf, goat and cabbage problem with NuSMV and use the counterexample-finding functionality to find a solution.

Hint: write and invariant that, if satisfied, corresponds to the property "there is no solution to the problem". What happens if you try to verify this invariant with NuSMV?

[1] Wikipedia, Wolf, goat and cabbage problem. `https://en.wikipedia.org/wiki/Wolf,_goat_and_cabbage_problem`