Spin: Exercises - Part B*

Patrick Trentin
patrick.trentin@unitn.it
http://disi.unitn.it/trentin

Formal Methods Lab Class, April 18, 2018

Università degli Studi di Trento

(compiled on 17/05/2018 at 12:53)

*These slides are derived from those by Stefano Tonetta, Alberto Griggio, Silvia Tomasi, Thi Thieu Hoa Le, Alessandra Giordani, Patrick Trentin for FM lab 2005/18

Exercise 1: Elaaden Vault

Exercise: Five ControlPillars, numbered from 0 to 4, control the gate of an ancient vault. Initially, pillars 1, 3 and 4 are in `ON` state, while 0 and 2 are *OFF*. The gate opens when all pillars are contemporarily set to `ON`.

- Each **ControlPillar** waits for input commands sent through their input channel `ctl`. Whenever a pillar receives a command, it **atomically** changes its own state –and the state of its immediate left and right neighbours– to the opposite value. To this extent, pillars 0 and 4 must be considered neighbours of each other.
- A spaceship **Commander** keeps sending command messages to randomly chosen control pillars, up until the gate opens.

Write a property `p1` s.t. its counter-example is a sequence of button-switches that will open the gate.

Q: how can **Spin** be used to find the **minimum-length** sequence that opens the gate?

Exercise 2: oscillator

Exercise: Write a Promela model that initializes a global integer variable **sum** to be 0. Model a process **P**, stuck in an infinite loop, which:

- draws a random value included in $\{1, 3\}$ and assigns it to **v**
- updates the value of **sum** as follows:
  - if **sum** is positive valued, it subtracts **v** to its value
  - otherwise, it adds **v** to its value

Verify the following **ltl** properties:

- the value of **sum** is equal to 0 infinitely often
- the value of **sum** is never larger than 3 or smaller than -3
- it always the case that if **sum** is greater than 0 then it will eventually be smaller than 0, and if **sum** is smaller than 0 then it will eventually be larger than 0

Q: why is the third property **not** verified? can you fix it?

Exercise 3: cigarette smokers

Exercise: Assume that a cigarette requires three ingredients to be made: `TOBACCO`, `PAPER` and `MATCHES`. There are three smokers around a table, each of which has an infinite supply of only **one** ingredient.

- **Smoker.** Each smoker is in a loop waiting for both of his missing ingredients to appear on the *table*. Whenever that happens, he grabs the ingredients (the table becomes empty), rolls a cigarette and smokes it by printing a message. A smoker must also put one unit of his own resource on the table whenever asked to do so.
- **Master Agent.** Whenever the table is empty, the *master agent* sends a message demanding a unit of resource to be put on the table to two distinct *smokers* using a channel. The *master agent* chooses the *smokers* that have to put their own resource on the table using a *uniform random distribution*.

Simulate the system and visually verify that it behaves correctly: the simulation output consists of an infinite execution trace in which each **smoker** smokes infinitely often.

Exercise 4: railway station

Exercise: In a railway station **trains** are countinuously arriving and leaving. Goods are contained in some cargos and, depending on the weight, they are moved from/to either **trucks** or **vans**.
Write a Promela program that models this scenario considering **each cargo as a message** that should be sent/received through the right channel. Each **channel** (train, truck and van) can contain **16 cargos** as a maximum. The **maximum weight** of each cargo in a van is **128**.
You will need two processes:

- ‘‘`split`’’, that splits goods from the train channel, dividing them over the other two channels, truck and van, depending on the weight values attached
- ‘‘`merge`’’, that merges the two streams back into one, most likely in a different order, and writes it back into the train channel.

Here are the initial cargo weights on the train: $345, 12, 6777, 32, 0$;

Example 5: word counter

Exercise: In each sentence (string hereafter) the number of the characters composing the string is greater or equal than the number of the words contained in the sentence. A word is characterized by delimiters:

- space ' '
- tabulation '\t'
- endline '\n'

Write a spin function **count()** that perfoms property-based slicing of a string channel, counts the number of characters **nc** and the number of words **nw** and checks if the property $nc >= nw$ is always true.

Use the init function to pass to count() a string (remember that you can model a string as a channel of integers corresponding to ASCII characters).

Exercises Solutions

- will be uploaded on course website within a couple of days
- send me an email if you need help or you just want to propose your own solution for a review

- learning programming languages requires practice: try to come up with your own solutions first!