



Universidade de São Paulo

Programa de Pós-Graduação em Engenharia Mecânica

PMR 5237 – Modelagem e Design de Sist. Discretos em Redes de Petri

5a. Lista de exercícios

Prof. José Reinaldo Silva

Nesta que é a nossa última lista de exercícios vamos tartar com a modelagem e design de sistemas diretamente e não apenas com as técnicas que podemos usar para isso. Assim, considere o seguinte problema:

A driving school is trying to set up an information system to track the progress of the students training and the deployment of instructors. As a starting point for a formal process model the following description can be used.

New students register with the driving school. A registered student takes one or more driving lessons, followed by an examination. Each driving lesson has a beginning and an end. Instructors give driving lessons. The driving school has five instructors. Each driving lesson is followed by either another lesson or an examination. The examination has a beginning and an end and is supervised by a driving examiner. In total there are ten driving examiners. For the outcome of an examination there are three possibilities:

1. The student passes and leaves the driving school.
2. The student fails and takes additional lessons in order to try again.
3. The student fails and gives up.

Use a colored Petri net to model that one takes ten lessons before taking the exam and people will drop out if they fail three times.

Every token in the places *begin*, *c1*, *c2*, *c3*, *c4*, *c5*, *c5*, *end* has a value now. For instance: A person named J. Walker, 18 years old who has taken no lessons and no exams yet is represented as:

```
[id: 'X07'; name: 'J. Walker'; age: '18'; gender: 'male'; nof_lessons: "0"; nof_exams: "0"]
```

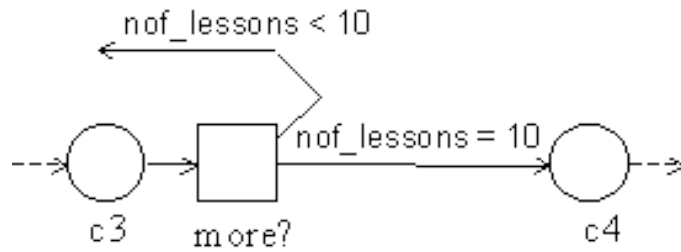
The last two attributes are important to the exercise, because we want two know how many lessons and exams a person already have had.

The transitions are specified as follows:

```
register: nof_lessons: = 0  
nof_exams: = 0
```

The transition to *more* and *ready* can be fused in one transition: *more?* with the following behavior:

if
nof_lessons < 10
then
produce token for c1
else
produce token for c4



***end_lesson*: nof_lessons: = nof_lessons + 1**
***end_exams*: nof_exams: = nof_exams + 1**
***again* has a precondition: nof_exams < 3**
set the attribute nof_lessons: = 0, because one has to take another ten lessons before the next exam.