

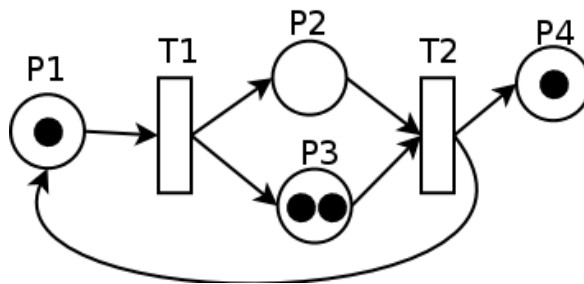
TP

Exercice 1

Un réseau de Petri se représente par un graphe biparti (composé de deux types de nœuds et dont aucun arc ne relie deux nœuds de même type) orienté (composé d'arc(s) ayant un sens) reliant des places et des transitions (les nœuds). Deux places ne peuvent pas être reliées entre elles, ni deux transitions. Les places peuvent contenir des jetons, représentant généralement des ressources disponibles.

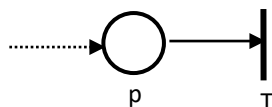
La distribution des jetons dans les places est appelée le marquage du réseau de Petri.

Les entrées d'une transition sont les places desquelles part une flèche pointant vers cette transition, et les sorties d'une transition sont les places pointées par une flèche ayant pour origine cette transition.



Travail demandé

1. Créez un méta-modèle ecore pour les réseaux de Petri.
2. Créez un méta-modèle ecore pour les graphes
3. Ecrivez un programme Java qui permet de transformer un modèle RdP (passée en paramètre) en un graphe.
4. Ecrivez une contrainte OCL qui permet de vérifier que pour chaque transition dans un RdP, il y a au moins un arc en entrée et au moins un arc en sortie.
5. Une transition T d'un RdP est **vivante** si elle peut toujours être franchie à partir d'un marquage atteignable quelconque. Plus généralement, on dira qu'un **RdP est vivant** si toutes ses transitions sont vivantes.
 - a- Ecrivez une contrainte OCL dans le **context Transition** qui permet de vérifier que la transition est vivante.
 - b- Ecrivez une contrainte OCL dans le **context RdP** qui permet de vérifier que au moins 50% des transitions d'un RdP sont vivantes.
 - c- Ecrivez une contrainte OCL dans le **context RdP** qui permet de vérifier que le RdP est vivant.
6. Une transition **puits** est une transition qui ne comporte aucune place de sortie (T est une transition puits).



Ecrivez une contrainte OCL qui vérifie que le RdP contient au moins une **transition puits**.