

T.P.1 RPHN (Boite à Outils TINA)

Introduction

La boite à outil Tina (<http://www.laas.fr/tina/>) propose plusieurs outils permettant de modéliser, simuler et vérifier des réseaux de Petri. L'objectif de ce TP est de vous familiariser avec ces outils. N'hésitez pas à lire la documentation et tester l'éditeur sur des exemples simples avant de vous lancer dans la réalisation des exercices.

Exercice 1 :

Considérer l'algorithme d'exclusion mutuelle vu en cours.

1. Dessiner le réseau de Petri correspondant
2. Lancer le simulateur pas à pas (onglet **tools/stepper**)
3. Simuler le fonctionnement du système. Trouver le comportement menant vers un état de blocage.
4. Construire le graphe des marquages et regarder le fichier résultat.
5. Exprimer et vérifier la propriété d'équité
6. exprimer et vérifier la propriété d'absence de blocage

Exercice 2 : Feux de Circulation

1. Modéliser à l'aide d'un réseau de Petri l'activité cyclique d'un feu tricolore de circulation.
2. Représenter le fonctionnement de deux feux coordonnant la circulation à l'intersection de deux boulevards. Un feu de circulation ne peut être vert que si l'autre feu se trouve au rouge. On considère initialement que le premier feu est rouge alors que le second est vert.
3. Dessinez le graphe des états accessibles.
4. Exprimez et vérifiez les propriétés suivantes (en cas d'une réponse négative donnez le contre exemple fourni) :
 - (a) Les deux feux ne sont jamais verts en même temps.
 - (b) Quand un feu est au rouge, il passera au vert un jour.

Exercice 3 : Lecteurs Ecrivains

La Figure ci-dessous représente une modélisation d'un problème d'accès concurrent à une base de donnée par 6 lecteurs et 3 rédacteurs. Pour assurer une certaine cohérence des données de la base, il faut interdire l'accès (en lecture et en écriture) à tous les processus, si un processus est en train de modifier la base (accède à la base en mode écriture). Par contre, plusieurs processus peuvent accéder à la base, en même temps, en mode lecture. Les rédacteurs (writers) représentent les processus qui demandent des accès en écriture à la base de données. Les lecteurs (readers) représentent les processus qui demandent des accès en lecture à la base de données.

1. Construisez le graphe des marquages et renseignez le nombre d'états et d'arcs.
2. Exprimez et vérifiez les propriétés suivantes (en cas d'une réponse négative donnez le contre exemple fourni)
 - (a) La place lock est 5 bornée
 - (b) Lecture et Ecriture sont exclusives

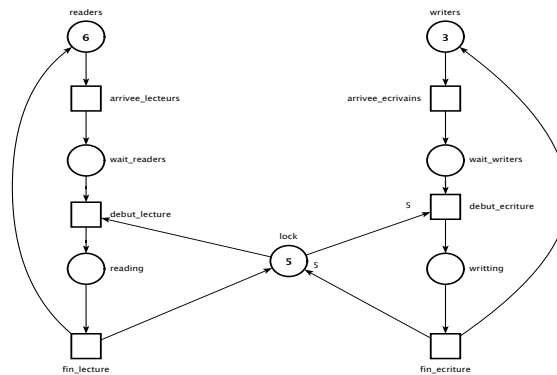


FIGURE 1 – Problème des lecteurs/écrivains

- (c) Au maximum 5 lectures possibles en parallèle
- (d) Toute arrivée de lecteurs sera suivie par une lecture
- (e) Tout début de lecture sera suivi par une fin de lecture

Exercice 3 : Problème du coiffeur

Modéliser avec un réseau de Petri un salon de coiffure en considérant la description suivante :

- Le salon dispose d’une salle d’attente de capacité n ,
- Le coiffeur est dans deux états possibles : **occupé** et **libre**
- Quand un client rentre dans le salon, il a trois comportements possibles :
 - Si le coiffeur est libre et la salle d’attente est vide, il se fait coiffer. Le coiffeur passe alors à l’état **occupé**,
 - Si il y a une chaise libre, il prend une chaise et s’installe en salle d’attente,
 - Si la salle d’attente est pleine, il quitte le salon.

Dans la suite, nous considérons que la salle d’attente est de capacité 2.

1. Lancer le simulateur et vérifier que votre modélisation satisfait la spécification.
2. Est-ce que le modèle est borné? Si non, simuler un comportement faisant indéfiniment augmenter le marquage d’une place donnée.
3. Construire le graphe de couverture du réseau et visualiser le fichier résultat.
4. Modifier votre modélisation de manière à rendre le réseau borné (penser à modéliser un SAS à l’entrée du salon).
5. Construire le graphe des marquage et visualiser le fichier résultat.
6. Imaginer, exprimer et vérifier des propriétés (trois au moins) qui vous semblent pertinentes concernant ce système.