

Exercices TLA

8 février 2011

1 Programmes

1.1 Factorielle

1. Écrire une fonction *définissant* la factorielle de x .
2. Écrire un programme qui *calcule* la factorielle de x :
 - itérativement, par multiplications/décrémentations successives ;
 - non déterministiquement, en construisant itérativement le produit des entiers entre 1 et x (deux versions : avec ensemble, avec tableau)

1.2 Pgcd

1. Écrire une (des) définitions du pgcd de deux nombres.
2. Écrire un programme qui calcule le pgcd par l'algorithme d'Euclide.

1.3 xyplus1

1. Écrire un programme constitué de deux variables x et y , avec deux actions, l'un qui transfère la valeur de x dans y , l'autre qui transfère la valeur de $y + 1$ dans x .
2. Formuler les propriétés (intuitivement) vérifiées par ce programme.
3. Prouver formellement la propriété d'invariance.
4. Prouver formellement la propriété de vivacité.

2 Résolution de problèmes

2.1 L'homme, le loup, le mouton et le chou

Sur une rive, se trouvent un Homme, un Loup, un Mouton et un (gros) Chou. Une barque permet de traverser, mais elle ne possède que deux places, et seul l'Homme peut ramer.

Seul la présence de l'Homme peut assurer une coexistence pacifique entre le Loup et le Mouton d'une part, et entre le Mouton et le Chou d'autre part.

Modéliser ce problème, avec la *négation* du résultat attendu comme propriété de sûreté, de manière que son *invalidation* démontre l'existence d'une solution.

2.2 Les missionnaires et les cannibales

Trois missionnaires et trois cannibales sont sur la rive gauche d'un fleuve. Ils disposent d'une barque qui peut transporter deux personnes (rameur inclus). Si les cannibales se retrouvent en majorité absolue sur une rive, ils mangent les missionnaires. Les six peuvent-ils traverser le fleuve en sécurité (pour les missionnaires) ?

Même problème avec quatre missionnaires et quatre cannibales.

3 Parallélisme

3.1 Exclusion mutuelle : Peterson

1. Modéliser le comportement des processus demandant et libérant l'exclusion mutuelle.
2. Spécifier les propriétés attendues par un « bon » protocole d'exclusion mutuelle.
3. Coder l'algorithme de Peterson.
4. Validation :
 - (a) Construire le graphe des états accessibles ;
 - (b) Vérifier les propriétés de sûreté ;
 - (c) Intuitivement, quelles propriétés d'équité faut-il imposer pour obtenir la vivacité ?
 - (d) Vérifier l'absence de famine.

3.2 Lecteurs/rédacteurs

1. Modéliser le problème des lecteurs/rédacteurs. Indication : utiliser deux compteurs, pour compter le nombre de lecteurs en cours et le nombre de rédacteurs en cours.
2. Modéliser la stratégie priorité aux rédacteurs.
3. Démontrer que cette deuxième implantation *raffine* la première.
4. Modéliser la stratégie improprement nommée « lecteurs coalisés ».
5. Vérifier l'absence de famine.
6. Démontrer que cette troisième implantation *raffine* la deuxième.

3.3 Exclusion mutuelle avec jeton circulant

But : décrire l'exclusion mutuelle avec circulation d'un jeton sur un anneau.

Objectif : obtenir un code prouvé et implantable sur un système réparti sans mémoire commune.

3.4 Philosophes

Étudier le problème des philosophes :

1. Solution en utilisant l'état des voisins. Étudier l'équité.
2. Introduire les fourchettes. Un philosophe doit avoir les deux fourchettes voisines, dans n'importe quel ordre. Démontrer la présence d'interblocage.
3. Ordonner la prise des fourchettes : tous droitiers, sauf un gaucher. Étudier l'absence de famine et l'équité nécessaire.
4. Proposer une solution sans famine et sans équité forte.