

**Question 1.4.** La variable *witness* est supposée contenir la somme des entiers de 0 à  $N\%4$ . La vérification avec l'outil produit une erreur:

```
/\ b = 3
/\ canal = <<>>
/\ i = 4
/\ mes = -1
/\ pc = <<"Done", "Done", "Done">>
/\ result = 24
/\ s = 10
/\ s_ = 24
/\ witness = 40
/\ x = 13
```

Expliquer d'où vient cette erreur et apporter une correction si nécessaire aux processus proposés.

**Exercice 2.** Soit le petit module *qquestion3a.tla*.

Donner les deux expressions  $A1$  et  $A2$  à placer dans les parties *assert* afin que la vérification ne détecte pas d'erreurs dans cette assertion. Par exemple, on pourrait proposer  $(x = 1 \vee x = 2) \wedge (y = 0 \vee y = 5)$  mais il vous appartient de simuler le programme pluscal pour vérifier que jamais l'assertion que vous proposerez ne soit fausse. La solution *TRUE* fonctionne mais n'est pas autorisée et les expressions demandées doivent contenir une occurrence de  $x$  au moins et une occurrence de  $y$ .

**Exercice 3.** On considère des populations de clients  $P_i = \{P_{ij} : j \in 1..n_i\}$  avec  $i \in 1..n$  et  $Q_j, j \in 1..n$  associé à chaque population: le processus  $Q_i$  est le serveur de la population  $P_i$ . L'algorithme de la figure 4 met en place la gestion d'une ressource  $R$  partagée par les processus  $C_1 \dots C_p$  via un serveur  $S$ . On décide d'utiliser cet algorithme pour gérer une ressource  $R$  partagée par toutes les populations et attribuée aux populations par leur serveur respectif quand ce serveur a le jeton. Le réseau en anneau dans la figure 3 explique les liens possibles entre les processus serveurs  $Q_i$  et les populations. La gestion de l'anneau est réalisé comme indiqué dans le fichier *qring.tla* de la figure 3. La gestion d'une population est assurée par le programme du fichier de la figure 4.

**Question 3.1.** On souhaite tester le protocole ring du fichier *qring.tla* de la figure 3. Expliquer pourquoi le réseau ring de *qring.tla* est correct.

**Question 3.2.** La ressource  $R$  ne peut être attribuée que par un processus serveur  $Q$  qui a le jeton c'est-à-dire que  $v[Q] = \text{TOKEN}$  sinon *NIL*.

Modifier le processus  $Q$  afin de réaliser cette fonctionnalité d'attribution de la ressource  $R$  au processus de la population gérée par  $Q$  et répondant à une demande de la population selon le protocole de la figure 4.

**Question 3.3.** Détailler les propriétés de correction que doit satisfaire ce protocole.

**Exercice 4.** (Exclusion mutuelle)

Lr protocole de demande d'exclusion mutuelle proposé par Ricart et Agrawala améliore celui de Lamport connu sous le nom de Bakery.

**Question 4.1.** Expliquez pourquoi l'algorithme de Ricart-Agrawala améliore celui de Lamport.

**Question 4.2.** Donner la propriété de sûreté qui doit être vérifiée par l'algorithme de Ricart-Agrawala.