SDR

Jetons

Mutex par jetons – Exercices

Olivier Lemer



Ricart et Agrawala

Exercices

Exercice 1

Suppositions :

- Un délai de transit de 2s existe pour toute communication.
- Tout autre délai est relativement négligeable (e.g. traitement, copie, etc.)
- Une SC dure 5s.

- Les événements simultanés sont priorisés dans l'ordre suivant :
  - demande de SC, sortie de SC, gestion de message
  - le numéro d'émetteur sert à trier deux messages reçus simultanément

- Deux processus dans le système, et initialement
  - `A` a un timestamp à `2`,
  - `B` a un timestamp à `10`.

`A` demandera la SC au bout d'une seconde, et `B` au bout de 4s.

Exercice 2

Suppositions:

- Un délai de transit de 2s existe pour toute communication.
- Tout autre délai est relativement négligeable (e.g. traitement, copie, etc.)
- Une SC dure 5s.
- Les événements simultanés sont priorisés dans l'ordre suivant :
  - demande de SC, sortie de SC, gestion de message
  - le numéro d'émetteur sert à trier deux messages reçus simultanément

Trois processus, et initialement

- `A` a un timestamp à 10,
- `B` a un timestamp à 6,
- `C` a un timestamp à 4.

Demandes de SC

- `A` au bout de 1 seconde,
- C au bout de 2 secondes.

Exercice 3

Voyez-vous une opportunité d'amélioration ?
Un scénario dans lequel on envoie plus de messages que nécessaire.

Carvalho et Roucairol

Exercices

Exercice

Suppositions:

- Un délai de transit de 2s existe pour toute communication.
- Tout autre délai est relativement négligeable (e.g. traitement, copie, etc.)
- Une SC dure 5s.
- Les événements simultanés sont priorisés dans l'ordre suivant :
  - demande de SC, sortie de SC, gestion de message
  - le numéro d'émetteur sert à trier deux messages reçus simultanément

Trois processus dans le système, et initialement

- `A` a un timestamp à `5`, et possède les jetons de `B` et `C`,
- `B` a un timestamp à `2`, et possède le jeton de `C`,
- `C` a un timestamp à `8`.

Les demandes d'entrée en section critiques sont faites :

- par `A` : au bout de 4s
- par `B` : au bout de 1s
- par `C` : au bout de 4s