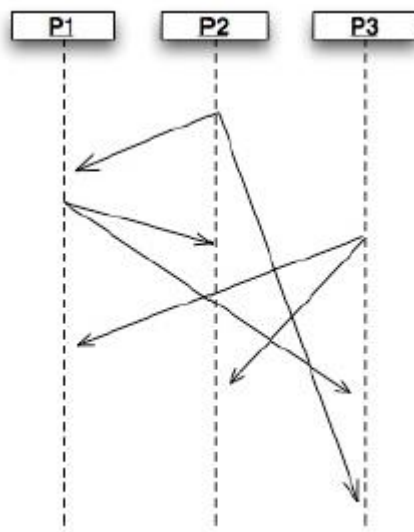


SERIE de TD N° : 02
MODULE : ALGORITHMES DISTRIBUES

Exercice 01 :

1. Appliquez l'algorithme de Ricart et Agrawala sur le diagramme suivant (les flèches représentent les messages du type requête($last_i, i$) et déterminez l'ordre d'entrée des processus en section critique.
2. Montrez qu'il n'y a jamais de famine.
3. Que doit-on faire si un processus tombe en panne après qu'un autre lui ait demandé la permission ?



Exercice 02:

On considère l'algorithme de Carvalho et Roucairol vu en cours.

Question : Donner une exécution où 3 sites veulent accéder à la ressource critique avec un site qui réclame deux fois de suite une ressource.

Exercice 03:

On considère le problème de la k-exclusion mutuelle : une ressource qui doit être accédée en exclusion mutuelle est disponible en k exemplaires. Toutes les demandes des processus ne portent que sur un exemplaire de la ressource et un processus n'émet pas de nouvelle

demande tant que sa demande en cours n'a pas été satisfaite. On peut donc avoir jusqu'à k processus simultanément en section critique.

Nous allons nous inspirer de l'algorithme proposé en cours (proposé par Ricart & Agrawala en 1983).

Q1: Combien d'autorisations faut-il recevoir pour garantir qu'au moins une ressource critique est libre ?

Q2 : Proposer une extension de l'algorithme proposé en cours.
