Manuel de l’utilisateur

Projet système d’exploitation

Le but de ce projet est de simuler des cas de section critique. Ces cas sont présentés en sortie du programme, en fonction d’un fichier entré par l’utilisateur, qui contient les sémaphores et leurs initialisations, ainsi que des instructions.

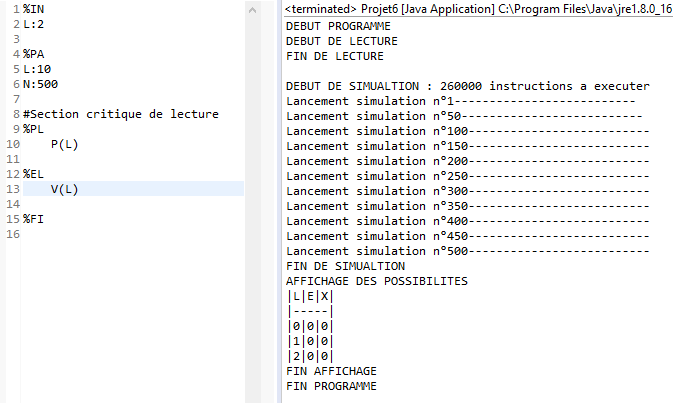
Carla MARTIN

Alexandre PACHOUD

Comment se servir de ce programme ?

L’utilisateur doit créer un fichier test comme décrit dans le sujet qu’il passe en argument. Ce fichier doit impérativement contenir les sections %IN et %FI. Il doit y avoir autant d’instructions P que d’instructions V pour une même section critique, sinon le programme affiche un message d’erreur.

Les résultats sont retournés dans un tableau comme sur la capture d’écran suivante :



Lancer le programme avec cette commande :

*$ java Semaphore filename.txt*

Où *filename.txt* est un fichier comme décrit précédemment fourni par l’utilisateur et dans le même dossier que l’exécutable.

**LES RESULTATS :**

Les résultats sont donnés dans un tableau à 3 colonnes intitulées L, E et X pour les trois sections critiques lecture, écriture et exécution. Les lignes de ce tableau donnent les combinaisons possibles du nombre de processus tournant dans chacune des sections critiques. Dans l’exemple donné précédemment, les combinaisons possibles sont :

Jusqu’à 2 lectures

Aucune écriture

Aucune exécution

Si l’utilisateur demande un nombre de simulations élevé, le programme l’informe du nombre d’instructions qui vont être exécutées, car si le nombre est trop grand, le temps d’exécution peut être long. De plus le programme indique sa progression dès qu’il effectue dix pour cent des simulations.

Comprendre les alertes du programme

Ce programme à un fonctionnement qui se passe en deux temps, une partie Lecture, puis une partie de simulation de la solution.

Si le fichier entré par l’utilisateur est erroné, la simulation ne peut pas avoir dans certain cas lieu, ou sera vide dans d’autres.

Le programme déclare les erreurs suivantes :

* Mauvaise déclaration d’instruction
* Mauvaise déclaration d’initialisation ou de paramètres
* Toutes redéfinitions
* L’équilibre entre P(X) et V(X)
* La présence de %IN et %PA

La ligne a laquelle le problème se situe est systématiquement signalé à la fin du message d’erreur.

Le programme ne déclare pas les erreurs suivantes :

* L’utilisation d’un sémaphore non initialisé
  + Dans ce cas le compteur du sémaphore est 0 donc le processus est automatiquement bloqué, pour toujours.
* Les données en dehors d’un %XX, elles sont simplement ignorées.
* Les erreurs de logique
  + Certaines solutions, bien que sans erreurs de syntaxes, peuvent conduire aux faits que tous les processus soit bloqué. Ces cas de figures peuvent être très simple.

Dans ces cas, si des processus se retrouvent bloqué pour toujours, alors la simulation s’interrompra automatiquement, pour en relancer une autres, et ainsi de suite. A la fin du nombre de simulation indiqué par l’utilisateur, le programme affichera le nombre de simulation qui ont échouées. Ainsi l’utilisateur peut avoir une approximation de la fréquence des bugs de la solution proposée. Cependant le programmes n’indique pas la source des problèmes rencontrés.