

Лекция 6. Алгоритмы синхронизации

Алгоритм Петерсона

Входная секция -- установка флага готовности в 1 и передача хода следующему процессу. После -- ожидание своего хода., когда другой процесс готов ко входу в критическую секцию. В выходной секции флаг готовности сбрасывается.

Алгоритм Петерсона работает только для двух процессов. Для произвольного количества процессов используется Алгоритм Булочной.

Аппаратная поддержка взаимоисключений

Команды

- test-and-set. Команда, проверяющая значение некоторой переменной и одновременно (атомарно) присваивает ей новое значение.
- swap. Меняет значения двух переменных местами. Также атомарно.

Алгоритмы синхронизации снижают продуктивность системы так как есть цикл перед критической секцией, где процесс ждет своей очереди, который занимает процессор.

Алгоритмы синхронизации также могут приводить к тупиковым ситуация при неудачном выборе алгоритма планирования. Например, если в системе реализовано вытесняющее приоритетное планирование и есть два процесса с низким и высоким приоритетом соответственно. Тогда, если низкий зайдет в критическую секцию, а в это время высокий дойдет до цикла во входной секции и объявит о своей готовности, то возникнет тупиковая ситуация. Высокий будет бесконечно ожидать, пока низкий выйдет из критической секции, а низкий не сможет получить в свое распоряжение процессор, ведь его постоянно занимает высокоприоритетный процесс.

Недостатки программных алгоритмов

Недостатки алгоритмов происходят из-за того, что ожидающие процессы на самом деле находятся в состоянии “Готовность” или “Исполнение”, а не

“Ожидание”, как должны быть. При этом перевести процесс в состояние ожидания в программе нельзя, это может сделать лишь ОС.

Семафоры Дейкстры

Механизм семафоров был предложен в 1965 году Дейкстрой.

ОС способна по запросу процессов создавать разделяемые неотрицательные целочисленные переменные “семафоры”. Над этой переменной можно выполнять три операции:

- присвоение произвольного значения.
- если значение семафора равно нулю, то процесс переводится в ожидание до тех пор, пока семафор не станет больше нуля. После этого из s вычитается единица.
- значение семафора увеличивается на единицу.

ОС обеспечивает атомарность операций над семафорами.

Мониторы Хоара

В 1974 году Хоар предложил концепцию монитора Хоара.

Механизм более высокого уровня, по сравнению с семафорами. Реализация монитора осуществлена таким образом, чтобы в состоянии готовности или исполнения мог находиться только один процесс. У монитора есть условные переменные, над которыми можно произвести операции wait и signal.

- wait -- блокирование процесса, который выполнил эту операцию. Процесс будет находиться в заблокированном состоянии до тех пор, пока другой процесс не выполнит операцию signal
- signal -- процесс, который выполняет операцию signal, мгновенно выкидывается из монитора.

Очередь сообщений

С очередью можно выполнять операции передачи и приема сообщений. ОС обеспечивает взаимоисключение на время закладывания информации в очередь сообщений и сбора информации из нее. Она также включает в себя блокирование процесса при попытке записи сообщения в заполненную очередь и при попытке чтения из пустой очереди.

