Variáveis de Condição

Grupo de Sistemas Distribuídos
Universidade do Minho

1 Conceitos relevantes

- suspensão/retoma de execução dentro de zona crítica

2 Mecanismos

- variável de condição intrínseca em cada objecto
- métodos de Object: wait(), notify(), notifyAll()

3 Exercícios propostos

1. Implemente uma classe BoundedBuffer que ofereça as operações `void put(int v)` e `int get()` sobre um array cujo tamanho é definido no momento da construção de uma instância. O método `put()` deverá bloquear enquanto o array estiver cheio e o método `get()` deverá bloquear enquanto o array estiver vazio. Os métodos oferecidos podem estar sujeitas a invocações de threads concorrentes sobre uma instância partilhada. A classe BoundedBuffer deverá garantir a correcta execução em cenário multi-thread.

2. Considere um cenário produtor/consumidor sobre o BoundedBuffer do exercício anterior, com P produtores e C consumidores, com um número total de threads $C + P = N$ e tempos de produção e consumo T_p e T_c. Obtenha experimentalmente o número óptimo de threads de cada tipo a utilizar para maximizar o débito.

3. Implemente uma classe Barreira que ofereça um método `esperar()` cujo objectivo é garantir que cada thread que o invoque se bloqueie até que o número de threads nesta situação tenha atingido o valor N, passado ao construtor. Um objecto Barreira deverá poder ser reutilizado em sucessivas invocações de `esperar()` (e.g., para uma sincronização ao fim de cada uma de várias etapas de uma execução).