

## Capítulo 3

# Variáveis de Condição

### 3.1 Conceitos relevantes

- suspensão/retoma de execução dentro de zona crítica

### 3.2 Construções de linguagem

- mecanismo intrínseco de variável de condição em todos os objectos
- métodos `wait()`, `notify()`, `notifyAll()`

### 3.3 Exercícios propostos

- Reimplemente a classe `Banco` de modo a bloquear as operações que conduzam a saldos negativos.
- 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.
- 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 de uma sua instância.
- Implemente o mecanismo tradicional de semáforos à custa do mecanismo de variáveis de condição.