

# Exercício: Depósito de Caixas (problema do produtor/consumidor) com Wait-Notify

Este exercício é o problema clássico dos produtores e consumidores. Assuma que um produtor vai ao depósito armazenar as caixas que vai produzindo. Da mesma forma, um consumidor vai ao mesmo depósito para retirar caixas que vai consumir.

As seguintes atividades devem ser feitas:

- 1) Implemente uma classe `Deposito` que tenha os atributos: (i) a quantidade itens atualmente no depósito (inicialmente vamos assumir que o depósito possui uma caixa); e (ii) a capacidade máxima de itens no depósito (tem que ter esse parâmetro no construtor). E que tenha os seguintes métodos: (i) `armazenar`, que coloca uma caixa no depósito; (ii) `retirar`, que retira uma caixa do depósito; e (iii) `retornar` a quantidade itens atualmente no depósito. Obviamente que não é possível armazenar nenhuma caixa caso a capacidade máxima já tenha sido atingida. Da mesma forma, não é possível retirar nenhuma caixa, caso o depósito esteja vazio.
- 2) Implemente uma classe `Produtor` que deve funcionar como uma thread independente e que invoca o método `"armazenar"` do objeto depósito de forma a acrescentar caixas ao depósito. A classe `Produtor` deve receber no construtor uma referência para o objeto depósito, além de um inteiro correspondente ao tempo em milisegundos entre produções de caixas. No método `run`, execute somente uma única produção de uma caixa.
- 3) Implemente uma classe `Consumidor` que também funcione como uma thread independente e que invoca o método `"retirar"` do objeto depósito de forma a retirar caixas do depósito. A classe `Consumidor` deve receber no construtor uma referência para o mesmo objeto depósito e um inteiro correspondente ao tempo em milisegundos entre consumos de caixas. No método `run`, execute somente uma única retirada de uma caixa.
- 4) Utilize a estratégia `wait/notify` para controlar a produção e a retirada das caixas do depósito.
- 5) No método principal, defina uma `Thread Pool` com 20 threads, sendo que 10 será produtoras e 10 serão consumidoras.
- 6) No final da execução, imprima a quantidade de itens no depósito (que tem que ser exatamente igual a um).

Observações:

- 1) Use somente um arquivo para resolver esse problema;
- 2) Entregue somente o código Java.