



## Trabalho de Programação Concorrente Simulador de Supermercado

Suponha que, neste trabalho, você foi contratado para desenvolver um simulador de compras em um supermercado. Neste simulador, existirão alguns elementos a serem considerados:

- **Itens de compra:** poderão ser 10 itens diferentes (sabão, creme dental, salgadinho..... etc., cada um com um preço fixo no próprio simulador).
- **Comprador e seu carrinho de compra:** o comprador retira um carrinho na entrada do supermercado e começa as suas compras. Ele deverá comprar um número aleatório de itens (entre 10 e 30), sendo que cada item demora entre 1 e 2 segundos para ser selecionado e colocado no carrinho (simplificação). Após a seleção dos itens, o comprador se dirige ao caixa e começa a depositar os itens na esteira (1 segundo para depositar cada item). Ao final, ele recebe o valor total do caixa e paga a conta. Este elemento deverá ser modelado como uma *Thread*. Saídas exemplo:
  - Comprador: [hora:minuto:segundo] Iniciando compras.
  - Comprador: [hora:minuto:segundo] Colocando item X no carrinho
  - Comprador: [hora:minuto:segundo] Indo para o caixa
  - Comprador: [hora:minuto:segundo] Depositando item X na esteira
  - Comprador: [hora:minuto:segundo] Pagando conta de Z R\$
- **Esteira:** a esteira é onde o comprador coloca os itens e de onde caixa retira os itens. Este elemento deverá ser modelado como *buffer* compartilhado, similar ao desenvolvido no produtor/consumidor.
- **Caixa:** o caixa retira cada item da esteira e efetua a contabilização do mesmo, passando cada item para a rampa de ensacolamento. Cada item leva entre 2 a 4 segundos para ser processado. Ao final, ele indica ao comprador o valor a ser pago. Este elemento deverá ser modelado como uma *Thread*. Saídas exemplo:
  - Caixa: [hora:minuto:segundo] Passando item X
  - Caixa: [hora:minuto:segundo] Informando conta de Z R\$
- **Rampa de empacotamento:** é onde o caixa coloca os itens e o empacotador retira os itens. Este elemento deverá ser modelado como *buffer* compartilhado.
- **Empacotador:** o empacotador retira os itens da rampa de empacotador e os acomoda em sacolas (2 segundos para ensacar). Saídas exemplo:
  - Ensacolador: [hora:minuto:segundo] Acomodando item X na sacola
  - Ensacolador: [hora:minuto:segundo] Fim do empacotamento

A escolha dos mecanismos de sincronização ficará por conta do implementador. Todas as threads deverão ser ativadas no início do simulador, e deverão executar concorrentemente. Detalhes omitidos ficam de livre escolha do implementador. O sistema deverá simular a compra de um único comprador. Ao final, o *main* deverá imprimir:

-Fim da simulação: [hora:minuto:segundo]