Laboratório de Computação Paralela e Sistemas Distribuídos – Produtor / Consumidor

Autores: Frederico Oliveira Freitas
Luan Felipe Ribeiro Santos
Talles Souza Silva

1. Introdução

O objetivo deste trabalho prático consiste na aplicação dos conceitos do algoritmo Produtor/Consumidor tomado como base a produção/consumo de xícaras de café.

2. Implementação

Pacote Apresentação

Classe: Inicial

Classe que executa as threads produtor e consumidor

public class Inicial extends JFrame(): Definições de variáveis e componentes da tela inicial, bem como o layout que será retratado após a execução do código fonte, com a contagem do número de xícaras de cafés produzidas para o consumo.

public class mudaQuantidade extends Thread(): Invoca a thread responsável pela mudança da quantidade de xícaras de café produzidas..

public class Movimento extends Thread(): Invoca a Thread que realiza os movimentos da xícara de café na esteira durante a produção. Caso não consiga, lança uma exceção. Verifica se o estoque está cheio e se sim, fica parada.

Classe Main

public static void main(String[] args): Instancia uma nova Cesta definindo seu tamanho inicial, além de invocar a tela de funcionamento que será mostrada.
Realiza a chamada para executar as linhas de produção e de consumo das xícaras de café.
Instancia um array de produtores e outro de consumidores, a fim de controlar seus respectivos funcionamentos com o uso do synchronized e o controle a partir do notify().

Pacote Negócio

Classe Produtor

Classe que produz os itens e os coloca na cesta

public Produtor(Cesta cesta, int tamanhoCesta): Método construtor para a identificação de cada elemento da produção que é colocado na Cesta.

public void run(): Método que invoca a Thread implementada por Runnable e que é responsável pela produção das xícaras de café que são adicionados na cesta.
Faz a utilização do wait(); afim de aguardar se a produção está ocupada. Lança uma exceção caso não seja possível a espera.

public void init(): Método que dispara a execução da Thread.

.

Classe Cesta

Classe responsável por produzir os elementos que serão retirados pela parte consumidora.
Ela adiciona e sincroniza os elementos da produção e os dispõe para serem consumidos pelo Consumidor, retornando a quantidade disponível na cesta.

Classe Consumidor

Classe que consome os itens colocados na cesta

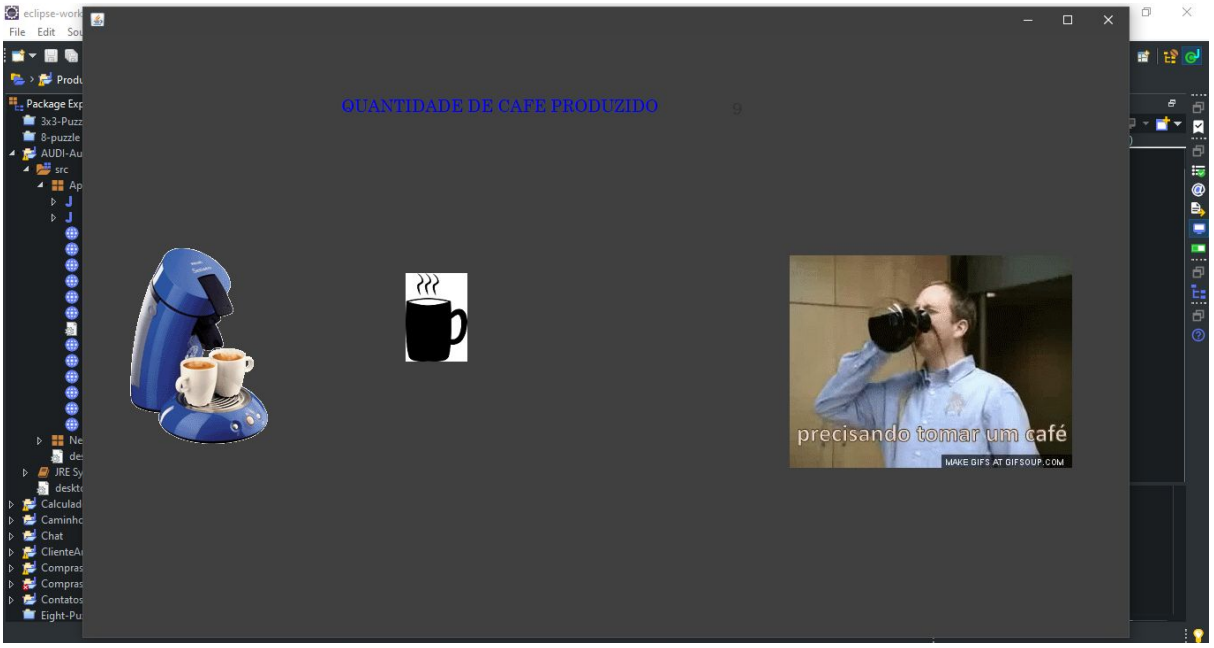
public Consumidor(Cesta cesta, int x): Método construtor para a identificação de cada elemento que é consumido da Cesta.

public void run(): Método que invoca a Thread implementada por Runnable e que é responsável pelo consumo das xícaras de café que são adicionados na cesta.
Faz a utilização do wait(); afim de aguardar se a produção está ocupada. Lança uma exceção caso não seja possível a espera.

public void init(int num): Método que dispara a execução da Thread.

3. Testes

- Interface gerada após a execução do código;
- Produção/Consumo de café.



4. Anexos

- Inicial.java
- Main.java
- Produtor.Java
- Consumidor.Java
- Cesta.java