

## Avaliação de Programação Paralela e Distribuída (PPD)

14 de dezembro de 2024

### Instruções

1. Leia atentamente a questão e desenvolva a sua solução adicionando o máximo de detalhes possível em sua resposta;
2. A organização da sua resposta é um aspecto importante e impactará no desempenho final desta avaliação;
3. Garanta que o software rode;
4. Esta avaliação possui 4 dias de duração; e
5. A nota desta avaliação está no intervalo entre (inclusive) 0 e 10.

Boa Prova!

### Implementação do Jantar dos Filósofos Distribuído

Com base no algoritmo do Problema do Jantar dos Filósofos e utilizando as interfaces `ServerSocket` e `Socket` da linguagem Java, desenvolva uma solução distribuída para este problema. Sua implementação deve atender aos seguintes requisitos:

1. **Cientes Filósofos:** Cada filósofo deve ser implementado como um cliente independente, isto é, cada um deverá estabelecer conexão com o servidor por meio de um `Socket`.
2. **Servidor Centralizado de Recursos:** Os recursos compartilhados (os "garfos") devem ser gerenciados por um servidor central. Esse servidor, implementado com `ServerSocket`, deverá aceitar conexões dos clientes (filósofos) e controlar o acesso aos recursos.
3. **Protocolo de Comunicação:** Defina um protocolo de comunicação claro e bem documentado entre servidor e clientes, inspirado em exemplos de RFCs (como SMTP ou FTP). Por exemplo, ao iniciar a conexão, o filósofo (cliente) pode enviar a mensagem `HELLO`, e o servidor responder `HI <código_único>`. Esse protocolo deverá permitir todo o fluxo de interação necessário, incluindo solicitações de acesso aos recursos, notificações de início de pensamento, solicitação para comer e devolução dos garfos.
4. **Identificação dos Filósofos:** Ao estabelecer a conexão inicial, o servidor deve atribuir um identificador (ID) único a cada filósofo. Em caso de perda de conexão, o filósofo deverá reconectar-se informando o seu ID prévio.
5. **Registros do Servidor:** O servidor deverá manter registros em memória sobre a atividade de cada filósofo, armazenando informações como a quantidade de vezes que o filósofo pensou e a quantidade de refeições realizadas.
6. **Tempo de Pensamento:** O tempo de pensamento de um filósofo deve ser um valor aleatório, em milissegundos, seguindo uma distribuição normal com média 5 e desvio padrão 2. Garanta que esse valor não seja negativo (valor mínimo zero) para manter a consistência do sistema.

### **Organização da Entrega:**

- **PROTOCOL.md:** Documento definindo o protocolo de comunicação entre filósofos (clientes) e servidor.
- **src/:** Diretório contendo todo o código-fonte da solução.
- **README.md:** Arquivo descrevendo como executar o código e detalhando a função dos principais componentes da solução.
- **ARCHITECTURE.drawio:** Diagrama de classes e componentes da solução (sugere-se usar o plugin Draw.io no VSCode).

### **Consultas Permitidas:**

É permitido consultar recursos online e discutir com colegas.

### **Proibição de Plágio:**

A cópia integral da solução de terceiros resultará em nota zero tanto para quem copiou quanto para quem forneceu a cópia. O ChatGPT e Gemini me ajudam nessa checagem, então, não faça.