

Trabalho Prático I

O grupo (max 3), deverá resolver um problema clássico da área de Sistemas Distribuídos, para tanto as instruções abaixo são brevemente elucidativas e, portanto, devem ser complementadas a partir da bibliografia da disciplina.

The readers and writers problem

O grupo deverá estudar o problema clássico de sincronização conhecido como "**the readers and writers problem**", problema dos leitores e escrevedores, aplicados a sistemas distribuídos. Para isto deve desenvolver um servidor de notícias que terá dois tipos de processos clientes: Os escritores, geradores de notícias (publishers); Leitores de notícias (subscribers);

Especificação:

O Servidor deve armazenar as notícias agrupadas por tópicos, que serão geradas e consumidas pelos escritores e apenas consumidas pelos leitores

Os escritores de notícias (publishers), "autenticados" deverão poder:

- Adicionar um tópico das notícias;
- Consultar tópicos existentes;
- Inserir uma notícia de um tópico;
- Consultar todas as notícias publicadas.

Os leitores de notícias (subscribers), " autenticados " podem:

- Se inscrever em um tópico (pode ser repetida em quantos tópicos interessar);
- Consultar notícias de um tópico por intervalo de datas.
- Consultar a última notícia de um tópico.

Pode existir leitores de notícias não "autenticados" que podem executar as operações.

- Consultar notícias de um tópico por intervalo de datas.
- Consultar a última notícia de um tópico.

A aplicação deverá ser desenvolvida com JAVA RMI. Sempre que um escritor (publishers) publicar uma notícia, essa notícia será enviada a todos os consumidores registados que estão inscritos nesse tópico (tratar situações em que o leitor não está conectado). O número de notícias para cada tópico deve ser limitado por um parâmetro. Quando o limite do número de notícias de um tópico é atingido, deve ser descartado a notícia mais antiga. A notícia tem um limite de 180 caracteres e um "*timestamp*" correspondente à data e hora de publicação no servidor. Os processos clientes (escritores e leitores) terão uma interface de texto com o usuário. Os processos clientes (escritores e leitores) deverão poder executar em máquinas diferentes da máquina do servidor.

O que deve ser entregue junto com o trabalho:

1. Apresentação em aula para o professor;
2. Redação de relatório técnico;
 - a. Introdução;
 - b. Descrição da arquitetura da aplicação;
 - c. Detalhes dos processos clientes e servidor;

- d. Documentação (tais como, diagramas de classes e estados);
- e. Descrição e apresentação dos algoritmos;
- 3. Código Fonte, o relatório deve conter uma seção com as explicações de como executar o projeto/fonte;
- 4. O Relatório deve conter um exemplo de aplicação dos problemas estudados.

Nota sobre atrasos:

Para cada semana de atraso é descontado um ponto do valor total do trabalho. Sendo assim, um trabalho com atraso de uma semana tem sua nota máxima igual a 9, dessa forma, o grupo que executou 50% receberá 50% de 9, logo 4,5. Sendo que o atraso máximo tolerado é de 3 semanas.

Boa Sorte!