



Exercícios de Sockets

1. Servidor de Fortunes: o objetivo deste exercício é imitar o funcionamento do conhecido biscoito da sorte chinês (*fortune cookie*) em um ambiente distribuído. O fortune, a cada invocação, imprime para o usuário uma frase escolhida aleatoriamente a partir de uma base de dados de frases.
 - A ideia nesse exercício é construir um servidor de fortunes que suporte 2 operações:
 - i. GET-FORTUNE: retorna uma frase correntemente armazenada no servidor, escolhida aleatoriamente.
 - ii. ADD-FORTUNE: adiciona uma no banco de frases do servidor.
 - iii. UPD-FORTUNE: modifica uma frase armazenada no servidor.
 - iv. LST-FORTUNE: lista todas as frases armazenadas.
 - O protocolo de comunicação entre clientes e servidor é baseado em comandos representados como texto. Assim ao receber a *string* "GET-FORTUNE" o servidor deve responder com uma frase. Por sua vez, a operação ADD-FORTUNE assume a forma: "ADD-FORTUNE <nova frase>" e UPD-FORTUNE altera a frase por posicionamento "UPD-FORTUNE <pos> <nova frase>".
2. Implemente um servidor que aceite a ligação de um cliente de cada vez. O servidor receberá de cada cliente, uma sequência de inteiros (pode optar tanto pelo formato binário como de texto) terminada com uma operação e, quando detectar a situação de *end of file* na *stream* de leitura do socket, inicia a operação no servidor. O resultado do servidor é o resultado da operação.
3. Implemente um jogo da forca remoto com 1 jogador. A definição da palavra a ser adivinhada ficará no servidor, sendo que ao início o servidor dará o tamanho da palavra (em caracteres). Faça a simulação em texto do jogo, retornando cada parte do boneco ou o caractere da palavra do servidor para o cliente.
4. Implemente um servidor de uma instituição financeira que permita a conexão de n clientes. Uma vez no sistema, o cliente poderá realizar depósito, saque, saldo ou sair da aplicação. É importante que dois ou mais terminais possam fazer alterações na mesma conta, com o mesmo cliente (ou não), tendo sua atualização on-line sem conflitos.
5. Construa um sistema que simule uma rede de lojas de departamento. Um sistema central deve receber dados de n filiais. O sistema de cada filial deverá simular a informação de compras do dia, contendo a sequência de compras e vendas efetuadas naquele dia. Faça uma simulação com, no mínimo 5 filiais, sendo que cada filial tenha uma movimentação diária de 1500 ocorrências (para que a simulação consiga ocorrer de forma mais real, use delays para simular um tempo maior para cada ocorrência).