



## Exercícios

1. Aplicação eco: O cliente lê do teclado e envia para o servidor. O servidor recebe e reenvia para o cliente. O cliente então imprime o que recebeu e reinicia a leitura do teclado.
2. 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 a frase correntemente armazenada no servidor
    - ii. SET-FORTUNE: modifica a frase armazenada no servidor.
  - 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 a frase corrente. Por sua vez, a operação SET-FORTUNE assume a forma: "SET-FORTUNE\nNova frase\n".
3. 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 – que terminará quando detectar a situação de *end of file* na *stream* de leitura do socket. No final da leitura dos valores inteiros, o servidor devolve ao cliente a soma correspondente.
4. Implemente um jogo da forca remoto, com apenas 1 jogador. A definição da palavra a ser adivinhada ficará no servidor. O jogador poderá ser desconectado do jogo se ele digitar a frase "bye" ou for enforcado. E pra entrar no jogo novamente, ele terá que aguardar uma nova conexão.
5. Implemente um servidor de um Banco, que permita a conexão de um cliente por vez. Dado o número da conta, o cliente poderá realizar depósito, saque, ver seu saldo ou sair da aplicação.
6. Reimplemente o servidor de Banco do exercício anterior de modo a que este aceite a conexão simultânea de múltiplos clientes.