

Trabalho Prático 3

1 Objetivos

O objetivo deste trabalho é se familiarizar com RPC (*Remote Procedure Call*) e RMI (*Remote Method Invocation*), aspectos fundamentais no projeto e desenvolvimento de sistemas distribuídos. Neste trabalho, você deve utilizar as linguagens de programação C e Java para implementar a tarefa abaixo, utilizando RPC e RMI, respectivamente.

Além da implementação, você deve testar seu programa executando os estudos de casos. Você deve preparar um relatório, com no máximo 4 páginas, com as decisões de projeto e implementação das funcionalidades especificadas, assim como a avaliação dos estudos de caso. Por fim, você deve preparar uma apresentação de no máximo 10 minutos sobre seu trabalho (como no relatório), a ser realizada no horário da aula. O trabalho pode ser individual ou em dupla.

2 Operações em vetores com RPC e RMI

Considere o problema de aplicar uma função sobre os elementos de um vetor. Considere ainda dois tipos de função, uma que opera e transforma cada elemento do vetor, e outra que retorna um valor que é função dos elementos do vetor. Exemplos do primeiro tipo:

1. Aplicar a função \log_b a cada elemento do vetor, para uma base $b > 1$.
2. Elevar cada elemento do vetor a uma potência $a > 0$.

Exemplos de função do segundo tipo:

1. Calcular a média aritmética dos valores do vetor.
2. Calcular o número de elementos do vetor que são maiores do um valor t .

Projete e implemente um serviço utilizando RPC e RMI que ofereça esta funcionalidade (tanto o cliente quanto o servidor devem estar na mesma linguagem de programação). Você deve implementar ao menos três funções de cada tipo acima, e que sejam parametrizadas. Por exemplo, a função potência onde o parâmetro é o valor da potência ao qual cada elemento do vetor será elevado: `int potencia_rpc(double a, double *vec, int vec_size)` eleva cada elemento do vetor `vec` que possui tamanho `vec_size` a potência `a`.

Projete e implemente um programa *multithreaded* que vai utilizar o serviço oferecido via RPC ou RMI. Seu programa deve alocar um vetor de N posições e inicializá-lo com valores aleatórios. Para agilizar o processamento, o programa deve utilizar K threads que farão uso do serviço RPC ou RMI em paralelo. Uma forma de dividir o problema é instruir cada thread para trabalhar com N/K posições contíguas do vetor, cada uma fazendo apenas uma chamada RPC ou RMI. Instrumente o tempo necessário para aplicar a função matemática no vetor (sem contar o tempo de inicialização).

Para o estudo de caso, considere $N = 10^8$ e $K = 1, 2, 4, 8, 16, 32, 64$, e utilize as diferentes funções que foram implementadas no serviço. Para cada valor de K e função matemática, obtenha o tempo de execução rodando o programa 20 vezes para calcular o tempo médio de execução (para cada combinação de valores, teremos um tempo médio de execução). Apresente um gráfico mostrando o tempo médio de execução em função do número de threads para cada valor de K (cada função matemática deve ser uma curva no gráfico). Faça a avaliação para o serviço implementado em RPC e RMI. Discuta os resultados obtidos.