

CO18103 - Programação Paralela

Remote Method Invocation (RMI) com Java

PROJETO FINAL

Prof. Dr. Marcos Amaris

Junho de 2023

Entrega **05/07/2023** - Entrega via SIGAA na atividade **Projeto Final**.

Descrição do Projeto

Em anexo a essa atividade está o código fonte de uma “versão enxuta” de uma aplicação de calculadora simples na linguagem de programação Java usando o **Terminal** como interface com o usuário e *Remote Method Interface* (**RMI**). Nessa aplicação há dois programas principais, cliente e servidor. O servidor distribui um conjunto de métodos através de processos servidores, e o programa cliente que busca métodos através de processos clientes. Cada equipe de projeto melhorará ambos programas:

- programa servidor.
- programa cliente.

O Servidor

O programa servidor implementará as interfaces para acessar aos métodos remotos. Escreva tendo em mente que poderão ocorrer várias execuções simultâneas do programa servidor: cada “processo servidor” (uma execução do programa servidor) implementará um repositório de métodos. Isto significa que o programa servidor deve receber como argumentos, na linha de comando, certos parâmetros que devem variar de um processo servidor para outro (o nome do servidor, por exemplo).

O cliente

O programa cliente será usado para exercitar o sistema. Ele deve permitir que o usuário:

- estabeleça uma conexão com um (processo) servidor;
- interaja com o repositório implementado pelo servidor:
 - examinando o nome do repositório e o numero de métodos nele contidas,
 - listando os métodos no repositório,
 - buscando um método (por código de método) no repositório com seus respectivos parâmetros.

Fique à vontade para definir como seu programa cliente vai fazer a interface com os usuários. O único requisito é que a interface com o usuário permita que o sistema seja exercitado da forma descrita acima.

Em particular, o programa cliente deve possibilitar que um usuário chame métodos de diferentes processos servidores que estão distribuídos por vários repositórios. Provavelmente o mais fácil é escrever um cliente com uma interface tipo linha de comando. Tal cliente apresentaria um *prompt* e ficaria esperando comandos do usuário.

O cliente usando “linha de comando” poderia funcionar. Tome-a como uma sugestão (incompleta, por sinal), que pode ser seguida ou não. Se você tiver gás para escrever um cliente com uma interface com o usuário mais elaborada e amigável (GUI), vá em frente!

Formato e condições da entrega

As instruções abaixo devem ser seguidas à risca:

1. O projeto deve ser feito em equipes de **máximo 4 estudantes**. Os grupos devem ser informados ao professor pelo SIGAA até no máximo dia **05/07/2023**. Podem se formar nos mesmos grupos dos seminários.
2. Dúvidas em relação ao projeto devem ser discutidas no fórum do **SIGAA**.
3. Todos são fortemente encorajados a participar das discussões e ajudar seus colegas.
4. Entregue junto com o código fonte do projeto um relatório em **PDF** detalhado descrevendo a solução implementada e alguns exemplos de uso da interface do projeto. Por exemplo, se você implementar um prompt de comando, cada exemplo consistiria de um sessão com os comandos necessários e suas respectivas saídas. Os cenários serão testados durante a correção, portanto assegure-se que eles estão completos e autoexplicativos.
5. A entrega deve ser feita no prazo indicado no início deste documento em um único arquivo contendo todo o código, relatório e arquivos relevantes via SIGAA em uma atividade que será criada exclusivamente para este fim.
6. Cuidado ao seguir tutoriais desatualizados na Internet. O arcabouço que implementa o Java RMI sofreu várias modificações ao longo dos anos.

Avaliação

A entrega do relatório em PDF e os códigos fontes são obrigatórios. Segue a forma de avaliação.

1. A “versão enxuta” em anexo está somente chamando o método **soma** de forma remota os outros estão sendo usados de forma local. Será exigido o mínimo de 7 métodos remotos.
2. Na “versão enxuta” em anexo o servidor e o cliente estão sem informações sobre IP e Porta aonde estão executando, por isso ele não está funcionando de forma distribuída. Conserte o servidor e cliente para receber como argumentos, na linha de comando, certos parâmetros que devem variar de um processo servidor para outro (IP, nome do servidor, Porta, por exemplo).
 - Grupo que entregar item 1 sem item 2, terá nota de 6 pontos.
 - Grupo que entregar o item 1 e 2, terá nota de 8 pontos.
 - Grupo que entregar o item 1 e item 2. Subirá 1 ponto a cada mais 9 métodos remotos adicionados.
3. O projeto não terá apresentação oral, é obrigatória a sua defesa presencial do que fez cada grupo no Laboratório 2 da Computação 1^{ra} andar no dia 5 de Julho no horário da disciplina.