

# Exercício de Avaliação IV

Java Remote Method Invocation

Data de entrega: **27 Maio 2019**

Sistemas Informáticos – MIEBIOM/MIEF

Departamento de Engenharia Informática



---

## Regras de submissão do trabalho:

- Este trabalho deve ser realizado por um **grupo de 2 alunos** e conta para a avaliação;
- Deve submeter um ficheiro **.zip** com o **código** (ficheiros de extensão .java) desenvolvido para **este exercício. Não use outro formato de compressão.**
- O nome do ficheiro **.zip** deve seguir o formato: **nome1-nome2.zip**;
- O aluno que submeter o trabalho tem **de associar o seu colega** de trabalho durante o processo de submissão;
- Após submeter o trabalho em inforestudante, tem de registar o **esforço despendido na realização do trabalho por cada aluno em horas gastas em aula e em horas extra-aula**. Deve contar o tempo desde o início do semestre, caso esteja a submeter o exercício I; caso contrário conta apenas o tempo desde a última submissão de exercício de avaliação. Para o efeito preencha o formulário disponível em:

<https://goo.gl/forms/tEjK3vUSt0iU3AC33>

---

## Descrição

Neste exercício pretende-se desenvolver uma versão distribuída do jogo N-em-linha, implementado no Exercício III (com algumas adições). Para o efeito, deve adaptar as duas aplicações criadas para responder ao exercício III, transformando-as num cliente e servidor RMI. O cliente poderá assim invocar um conjunto de operações implementadas no servidor RMI, que corresponderão à funcionalidade pedida para o jogo N-em-Linha e às quais adicionará as seguintes funcionalidades:

- **Poder suspender e continuar um jogo.** O utilizador pode suspender um certo jogo, terminar a aplicação cliente, tendo a possibilidade de continuar jogos suspensos mais tarde. Para tal, cada jogo deve ter um identificador e pode ser retomado fornecendo o identificador do mesmo. Deve haver um limite ao número de jogos suspensos.
- **Poder fazer *undo* de 3 jogadas.** O jogador pode fazer um total de 3 *undos* (desfazer a última ação no jogo) durante todo o jogo. Quando o jogador faz um *undo*, a jogada correspondente do computador (ou dos computadores, no caso do nível de dificuldade adicional) também deverá ser desfeita.
- **Poder terminar o jogo sem sair da aplicação.** O utilizador deve ter a possibilidade de terminar o jogo em curso sem sair da aplicação. Assim, pode iniciar um novo jogo ou, por exemplo, retomar um jogo entretanto suspenso.

Para manter os seus dados em disco, poderá usar uma *ObjectStream*<sup>1</sup>. Assim, o servidor ao iniciar deve ler todos os dados de disco para memória. Qualquer alteração ao que se encontre em memória deve resultar na atualização dos dados em disco.

Não é necessário realizar qualquer operação de autenticação do cliente no servidor. **A aplicação cliente deve ser tolerante a falhas temporárias no canal de comunicação.** Em caso de falha na invocação de uma operação remota, a aplicação deve aguardar X segundos e automaticamente voltar a tentar executar a operação que falhou. Se ao fim de N tentativas ainda não existir comunicação com o servidor, o cliente termina.

---

<sup>1</sup> Documentação disponível em

<https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/io/ObjectOutputStream.html> e  
<https://docs.oracle.com/en/java/javase/11/docs/api/java.base/java/io/ObjectInputStream.html>