

Trabalho Opcional 2

<u>Pontuação:</u>	1 ponto na Prova 1
<u>Grupo:</u>	Dois alunos (obrigatoriamente)
<u>Data de entrega:</u>	5 de Março
<u>Objetivo:</u>	Implementar cálculo de posições na série de Fibonacci com <i>threads</i> .
<u>Material a ser empregado:</u>	Java / Java Threads
<u>Material a ser entregue:</u>	Entregar, em meio digital, os seguintes itens <ul style="list-style-type: none">● Fonte das classes desenvolvidas;● Documentação dos fontes;● Medição de tempos de execução para diferentes posições da série: 10, 15, 20, 25, 30 (em uma tabela);● Exemplo de entrada com os respectivos resultados.<ul style="list-style-type: none">■ Importante: Não devem ser entregues arquivos em formato proprietário (ex. Word). Utilize formatos abertos, como arquivos texto ou "pdf".
<u>Avaliação:</u>	Os trabalhos serão apresentados por um integrante da equipe (sorteado pelo professor). A nota final para cada aluno do grupo será composta da seguinte forma: <ul style="list-style-type: none">● Nota da implementação: 1,0 ponto;● Nota da apresentação: fator entre 0 e 1,0. Este fator será utilizado para multiplicar a nota da implementação.<ul style="list-style-type: none">■ Importante: Uma vez sorteado o apresentador este não poderá mais contar com ajuda do grupo.
<u>Descrição:</u>	Deve ser implementado, com auxílio de threads em Java, o cálculo de posições em uma série de Fibonacci. Deve ser criada um thread para a computação de cada posição. Isto implica em criação recursiva de threads: a thread que calcula a posição n deve criar outras duas, uma para calcular $n-1$ e outra para calcular $n-2$ – isto enquanto $n > 2$.
<u>Dica:</u>	Posições elevadas na série, como $n=40$, podem necessitar um tempo de processamento muito elevado. Com $n>42$ também deve ser levado em conta os limites de representação dos tipos de dados de Java.