

Observações quanto à Avaliação:

- 1- A prova é individual e permite consulta, porém cópias da Internet ou dos colegas não são permitidos sob pena de zerar a prova.
- 2- A interpretação faz parte da avaliação.
- 3- Cada questão deve ser resolvida em um projeto Java próprio.
- 4- Critérios para a avaliação (quando cabível):
 - * sintaxe e semântica,
 - * nomenclatura (rotinas e variáveis),
 - * organização do código (espaçamento horizontal e vertical),
 - * tipos adequados para variáveis, parâmetros e retornos das rotinas,
 - * representação correta das equações no código,
 - * mensagens adequadas para a interação com o usuário do programa durante as leituras e as escritas.
 - * definição adequada das classes e instanciação dos objetos
- 5- Os arquivos gerados devem ser compactados em um arquivo .zip com o nome do aluno no formato NomeUltimoNome. Por exemplo: EdsonCamargo.zip

1. O objetivo deste exercício é compreender o conceito de *thread* e como elas podem ser usadas para utilizar melhor os recursos de processamento disponíveis nos computadores modernos. Para isso, será construído um programa em Java para calcular o valor de Π usando N *threads* simultâneas. O valor de Π pode ser aproximado pela série de Leibniz-Grégory:

$$\frac{\pi}{4} = \sum_{n=0}^{\infty} \frac{(-1)^n}{2n+1}$$

Exemplo para 5 termos: $\frac{\pi}{4} = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \frac{1}{9}$

Observações:

- Devem ser calculados de 500 milhões à 1 bilhão - 10^8 à 10^9 - de termos da série; use variáveis reais de dupla precisão (*double*) nos cálculos;
- O programa deve dividir o espaço de cálculo uniformemente entre as N *threads*; cada *thread* efetua uma soma parcial de forma autônoma;
- Para evitar o uso de mecanismos de sincronização, cada *thread* $T[i]$ deve depositar seu resultado parcial na posição *result[i]* de um vetor de resultados parciais. Após o término das *threads* de cálculo, o programa principal soma os resultados parciais obtidos por elas e apresenta o resultado final na tela;
- Para garantir um bom desempenho, evite operar muito com variáveis globais dentro das *threads*;
- Devem ser medidos os tempos de execução do programa para execuções com 1, 2, 4, 8 e 16 *threads*.
- Ao executar seus experimentos, verifique se a máquina não está muito carregada, o que pode falsear os resultados (use o comando *top* para ver a carga da máquina);
- Para que os resultados tenham valor estatístico, devem ser feitas pelo menos 3 medidas de cada experimento e calculados o tempo médio;

Entregar

- Deve ser entregue a uma versão Serial e outra versão Paralela (com *threads*).
- Crie uma classe para manipular a série e outra classe para realizar a execução. Veja o exemplo do Contador de Múltiplos visto em sala. Repare que no Contador de Múltiplos há uma versão onde o objeto não é compartilhado entre as *threads* e outra em que o objeto é compartilhado entre as *threads*.
- Assim como no contador de Múltiplos, entre uma versão onde o objeto não é compartilhado pelas *threads* e uma versão onde o objeto da classe para manipular a série é compartilhada entre todas as *threads*. Nessa última versão aplique uma solução para sincronização.
- Um gráfico em arquivo .pdf apresentando o tempo de execução para 1, 2, 4 e 8 e 16 *threads* para 10^8 ou 10^9 termos da série