

## Regras para a execução do trabalho:

1. O trabalho será entregue até o dia 19 de dezembro de 2022 e apresentado nos dias 19 ou 21 de dezembro de 2022. Trabalho atrasado será descontado 10% da nota máxima a cada dia de atraso.
2. O trabalho pode ser realizado em duplas e trios. Porém, será sorteado um aluno da dupla para apresentar o trabalho e tirar dúvidas sobre a implementação.
3. Um aluno, em nome da dupla ou trio, até o prazo de entrega, deverá fazer o envio **via Moodle** de um arquivo contendo o que se pede.

## Sobre a avaliação do trabalho:

1. A nota será composta pela qualidade técnica da solução desenvolvida. A nota é condicionada a apresentação em dia/horário previamente determinados.

## Definição do Trabalho

O trabalho consiste da implementação paralela de um algoritmo sequencial com o objetivo de reduzir o seu tempo de execução. Sendo assim, o trabalho está dividido nas seguintes etapas (para auxiliar na organização):

1. Definição do problema que será paralelizado. A equipe pode escolher qualquer problema disponível na página de problemas da maratona de programação paralela do WSCAD (exceto o código de warmup), conforme links abaixo:
  - <http://lspd.mackenzie.br/marathon/21/problems.html>
  - <http://lspd.mackenzie.br/marathon/20/problems.html>
  - <http://lspd.mackenzie.br/marathon/19/problems.html>
  - <http://lspd.mackenzie.br/marathon/18/problems.html>
  - <http://lspd.mackenzie.br/marathon/17/problems.html>
  - <http://lspd.mackenzie.br/marathon/16/problems.html>
  - <http://lspd.mackenzie.br/marathon/15/problems.html>
  - <http://lspd.mackenzie.br/marathon/14/problems.html>
  - <http://lspd.mackenzie.br/marathon/13/problems.html>
  - <http://lspd.mackenzie.br/marathon/12/problems.html>
  - <http://lspd.mackenzie.br/marathon/11/problems.html>
2. Para evitar duplicatas, assim que escolher o problema, o membro responsável pela equipe deve acessar o Fórum do moodle para descrever sua escolha.
3. Realizar a implementação do código paralelo em OpenMP. A equipe pode utilizar qualquer uma das diretivas vistas em sala de aula.

4. Realizar diversos testes de desempenho, variando o número de *threads* e o tamanho do conjunto de entrada. Para cada conjunto de entrada, gerar gráficos de *speedup* e eficiência de desempenho.
5. De posse dos gráficos, escrever um relatório informando as decisões tomadas durante a paralelização e justificando-as. Analisar também o desempenho da implementação paralela.