

#### Lab 4: Vazão e latencia de mensagens MPI Broadcast no cluster (parte a)

Histórico:

- v1.0 a versão inicial

-

Data do enunciado: 24/jan/2023

Data da entrega Lab4a: **02/fev/2023 (DATA determinada!)**

Objetivo do trabalho 4:

O objetivo deste exercício é fazer um trabalho inicial simples para prática de conceitos de MPI no cluster com rede ethernet. Ao mesmo tempo gostaríamos de ter uma idéia inicial das medidas de desempenho da troca de mensagens em MPI e se as mensagens de broadcast estão funcionando com multicast na rede ethernet.

O trabalho será dividido em:

- parte 4a (mais simples, reutilizando nossa implementação anterior de MPI ping pong síncrono, MAS fazendo funcionar os processos MPI em máquinas diferentes e se comunicando pela rede)
  - parte 4b (um pouco mais elaborada, tomando como base o 4ª, vamos trocar mensagens Send e receive ponto a ponto, por broadcast, deve funcionar para qualquer número np de processos, sendo que o número de processos é fornecido na linha de comando de execução do programa MPI.
  - Parte 4c (mais difícil um pouco, vamos tentar fazer uma implementação nossa de mpi broadcast. A nossa nova função deve tentar distribuir as mensagens para todos os outros nós a partir de um nó determinado como sendo o nó raiz. Devemos reaproveitar o programa da parte 4c, mas substituindo a chamada do MPI\_broadcast nativo pela chamada da NOSSA rotina MPI\_broadcast.
    - OBS: O nosso broadcast deve seguir a distribuição das mensagens fazendo uma “árvore” de envio de mensagens entre os nós. Devemos usar apenas sends e receives normais (ponto a ponto) do MPI. O aluno pode decidir se usará sends e receives síncronos ou assíncronos, o que for factível e mais eficiente possível.
    - A árvore de distribuição de mensagens entre os nós foi desenhada em sala (duas versões). Vamos escolher uma delas e fazer a implementação. Faremos novamente o desenho no quadro e possivelmente será incluído esse desenho nessa especificação (em anexo)
- 
- SOBRE AS EXPERIÊNCIAS:
    - Você deve medir 10 vezes cada experiência, para mensagens de tamanho:
      - 8 bytes, 1KB, 4KB, 16KB
    - Quantas mensagens deve enviar em cada experiência?
      - R: tente dimensionar o número de mensagens **nmsg** na linha de comando (para cada experiência) ou script, de modo que cada rodada (com um dado tamanho de mensagem) tome aproximadamente 1 segundo. Assim o tempo total para medir tudo não será muito alto (aproximadamente 40 segundos para todas as experiências bloqueantes e 40 segundos para NÃO bloqueantes).
  - SOBRE OS GRÁFICOS DAS EXPERIÊNCIAS:
    - Fazer um gráfico de barras da vazão média obtida para mensagens (eixo y) sendo o eixo x os tamanhos das mensagens: 8 bytes, 1KB, 4KB, 16KB

- Fazer outro gráfico de barras da latência média obtida para mensagens (eixo y) sendo o eixo x os tamanhos das mensagens: 8 bytes, 1KB, 4KB, 16KB
  - OBS: Faça os gráficos no mesmo tipo de planilha do trabalho 3 (mudando as tabelas e os calculos e os tipos dos gráficos)
- SOBRE O RELATÓRIO do trabalho:
  - Fazer um relatório (PDF) com suas conclusões. Descreva as características importantes da sua CPU usada. Inclua em apendice ao relatório o texto da saída do comando `lscpu` bem como a figura do comando `lstopo` para sua CPU usada.
  - OBS: será incluído nas páginas seguintes (EM BREVE) figuras com dicas de COMO fazer as medições e/ou distribuição de mensagens
  - Podemos discutir no lab inicialmente antes de bater o martelo!