

Exercício 1: Cálculo Distribuído da Média de um Array

Objetivo: Usar ``MPI_Scatter`` para distribuir partes de um array entre os processos, calcular a média local em cada processo e, em seguida, usar ``MPI_Gather`` para coletar as médias locais no processo raiz para calcular a média global.

Descrição:

1. O processo raiz inicializa um array com valores aleatórios e utiliza ``MPI_Scatter`` para distribuí-lo entre todos os processos.
2. Cada processo calcula a média de sua parte do array.
3. Todas as médias locais são coletadas pelo processo raiz usando ``MPI_Gather``.
4. O processo raiz calcula a média global das médias locais e imprime o resultado.

Exercício 2: Distribuição de Configuração para Cálculos Paralelos

Objetivo: Utilizar ``MPI_Bcast`` para enviar uma configuração inicial (por exemplo, o número de iterações para um cálculo) do processo raiz para todos os outros processos.

Descrição:

1. O processo raiz define uma variável de configuração, como o número de iterações para um cálculo intensivo.
2. Essa configuração é transmitida para todos os processos usando ``MPI_Bcast``.
3. Cada processo executa o cálculo com o número especificado de iterações.
4. Cada processo imprime seus resultados locais.

Exercício 3: Normalização de um Array em Paralelo

Objetivo: Normalizar um array grande em paralelo. O processo raiz calcula o valor máximo do array (**EXTRA: tente fazer esta parte do problema com OpenMP!**) e o transmite para todos os outros processos usando ``MPI_Bcast``. Cada processo então usa ``MPI_Scatter`` para receber seu segmento do array, normaliza-o, e os resultados são reunidos de volta usando ``MPI_Gather``.

Descrição:

1. O processo raiz cria um array com valores aleatórios e encontra o valor máximo.
2. O valor máximo é transmitido para todos os processos usando ``MPI_Bcast``.
3. O array é dividido e distribuído entre os processos com ``MPI_Scatter``.
4. Cada processo normaliza sua parte do array.
5. Os arrays normalizados são coletados no processo raiz usando ``MPI_Gather`` e o resultado final é impresso.

Exercício 4: Cálculo Distribuído de Desvio Padrão

Objetivo: Calcular o desvio padrão de um conjunto de dados distribuído. O processo raiz distribui o conjunto de dados usando ``MPI_Scatter``, cada processo calcula a média e a variação de sua parte do conjunto, e os resultados são coletados com ``MPI_Gather`` para calcular o desvio padrão global.

Descrição:

1. O processo raiz inicializa um array com valores aleatórios.
2. O array é distribuído entre todos os processos usando ``MPI_Scatter``.
3. Cada processo calcula a média e a variação de sua parte.
4. As médias e variações locais são coletadas no processo raiz usando ``MPI_Gather``.
5. O processo raiz calcula o desvio padrão global baseado nas médias e variações coletadas e imprime o resultado.