

UNIVERSIDADE FEDERAL DO RIO DE JANEIRO

Bacharelado em Ciência da Computação

Programação Paralela e Distribuída

Profº: Gabriel Pereira da Silva

# Relatório - Trabalho 1

Eduardo Barbosa - 116150432

Gabriel Villares - 114089936

# Questão 1

## Objetivo

A primeira questão tem como proposta a paralelização de 2 algoritmos, que calculam o valor de  $\pi$ , os mesmos fazem uso do método do trapézio (integral) e do método de monte carlo, respectivamente.

## Metodologia

Para ambos os algoritmos, a paralelização se deu em cima do loop “for”, uma vez que as operações que ocorriam dentro do mesmo eram independente entre as diversas iterações do loop. Deste modo, as iterações do “for” foram divididas para os  $n$  processos, onde o processo 0 realizam a iteração 0, o processo 1 a iteração 1 e assim sucessivamente, ou sejam, os processos realizam as iterações de modo intercalados.

Para a paralelização de ambos os algoritmo, foram necessário o uso do `MPI_Bcast` para difundir o valor de  $n$ , para todos os processos, e do `MPI_reduce` para somar os cálculos realizados por cada processo, nas diversas iterações do loop.

Vale ressaltar, que devido ao pedido de testes com  $n$  muito grandes ( $10^{10}$ ), foram necessárias alterações no código sequencial, para que os mesmos fossem capazes de realizar os cálculos de forma correta. A alteração consiste em mudar o tipo de  $i$ ,  $n$  e count de `int` para `unsigned long int`.

Os testes foram compostos pela a execução dos dois algoritmos com os valores de  $N$  igual a  $10^8$ ,  $10^9$ ,  $10^{10}$ , utilizando 1, 2, 4, 8, 16 e 32 processos. Por fim, para cada teste os algoritmos foram executado 3 vezes e a média do tempo de execução foi utilizada como resultado final.

## Conclusão

Após a realização dos testes e a geração dos gráficos, foi possível notar que no algoritmo que implementa o método do trapézio não houve redução no tempo de execução, de forma significativa, quando executado com mais de 4 processos. Tal observação é reafirmada ao se observar os gráficos de speedup e eficiência, onde fica evidente que a execução com 4 processos gera o melhor speedup e a execução com mais de 4 processo não trás uma eficiência melhor.

Para o algoritmo que implementa o método de Monte Carlo, foi notado que acima de 8 processos a redução do tempo de execução deixou de ser significativa, e da mesma forma foi possível reafirmar tal observação olhando para os gráficos de eficiência e de speedup.

Ao comparar ambos os métodos foi possível observar uma grande vantagem para o método do trapézio, onde o mesmo teve resultados de execução bem mais interessantes. Vale ressaltar que em ambos os métodos o speedup teve retorno negativo, o que mostra que o aumento de número de processos não trariam melhores resultados.