

- 1) Neste trabalho você deve paralelizar o programa **calpi**, que calcula o valor de π (3.14159265358979323846264 ...) usando uma aproximação do método de integral com o método do trapézio ($f(x)=4/(1+x^2)$) e com o método de Monte Carlo. É uma boa aplicação para se paralelizar porque a divisão do trabalho entre as tarefas é muito simples. Para isso observe os seguintes passos: **(2,0 pontos)**
 - Avalie o desempenho da versão paralela com OpenACC, com valores de N para 10^9 , 10^{10} e 10^{11} .
 - Trace gráficos com o tempo total de execução e speedup em relação à versão sequencial.
 - Apresente um relatório com código fonte, resultados e comentários sobre todo esse processo.
- 2) O segundo trabalho consiste em paralelizar o algoritmo de ordenação chamado RankSort ou Enumeration Sort: **(2,0 pontos)**
 - Avalie o desempenho da versão paralela com OpenACC, com valores de N para 10^6 , 10^7 e 10^8 .
 - Trace gráficos com o tempo total de execução e speedup em relação à versão sequencial.
 - Avalie a movimentação de dados e se há otimizações possíveis.
 - Apresente um relatório com código fonte, resultados e comentários sobre todo esse processo.
- 3) Neste trabalho você deve modificar um programa que calcula a equação de uma reta utilizando o método de mínimos quadrados. Antes disso, contudo, você deve escrever um pequeno programa que gere um arquivo cujo primeiro dado é um valor **N**, seguido de **N** pares de coordenadas **X** e **Y** que vão servir como entrada de dados. Você pode usar uma equação conhecida e atribuir um pequeno erro aleatoriamente para cada y gerado. Para modificar o programa observe as seguintes instruções: **(2,0 pontos)**.
 - Elabore uma versão OpenACC do programa do método dos mínimos quadrados que pode ser obtida no AVA.
 - Avalie o desempenho da versão paralela do programa com valores de N para 10^6 , 10^7 e 10^8 .
 - Trace gráficos com o tempo total de execução, speedup e eficiência.
 - Avalie a movimentação de dados e se há otimizações possíveis.
 - Apresente um relatório com código fonte, resultados e comentários sobre todo esse processo.
- 4) Neste trabalho você deve paralelizar o programa **mandelbrot**, que calcula um fractal de mandelbrot, utilizando rotinas do OpenACC. Ao paralelizar o código do **mandelbrot**, você aumentará sua compreensão dos modelos de programação, ganhará experiência no desenvolvimento de uma aplicação MPI. Para isso observe os seguintes passos: **(2,0 pontos)**.
 - Compare o desempenho com a versão sequencial. Avalie a movimentação de dados realizada na versão paralela e verifique se há otimizações possíveis. Apresente um relatório com código fonte, resultados e comentários sobre todo esse processo.
- 5) Escreva uma versão em OpenACC para o programa de cálculo de números primos do livro. **(2,0 pontos)**
 - Avalie o desempenho da versão paralela com OpenACC, Utilize valores de N igual a 10^9 , 10^{10} e 10^{11} .
 - Trace gráficos com o tempo total de execução e speedup em relação à versão sequencial.
 - Apresente um relatório com código fonte, resultados e comentários sobre todo esse processo.