

# **Desenvolvendo uma aplicação em BSP/CGM**

(Bulk Synchronous Parallel / Coarse Grained Multicomputer)

## **EP 3 – Entrega 02/12/2012**

Computação Paralela e Distribuída – MAC5742/MAC0431  
Segundo Semestre de 2012  
Prof. Alfredo Goldman (gold at ime.usp.br)

### **1. Introdução**

Essa atividade explora os conceitos e práticas da programação em BSP/CGM. BSP/CGM é considerado um modelo de programação. Enquanto o MPI é o arcabouço onde esse modelo é implementado e executado. Um dos objetivos do CGM é otimizar a comunicação que ocorre entre os nós processadores do modelo BSP. Com isso, é comum chamarmos de modelo de programação paralela BSP/CGM, mas de fato o arcabouço é o MPI. O mecanismo desse modelo consiste basicamente em dividir tarefas para os processadores ( $n/p$ ,  $n$  = tamanho da entrada e  $p$  = processadores) e criar barreiras de sincronização. O papel da barreira é fazer com que toda a comunicação seja finalizada antes de iniciar um próximo passo de computação. O resultado final do processamento é obtido após  $x$  passos.

Existem diversas versões de MPI, como openmpi, mpich, etc. Embora essas versões tenham a mesma natureza e objetivos, podem ocorrer divergências. Para essa atividade deve ser utilizada a versão mpich2 juntamente com as bibliotecas do cgmlib. O cgmlib pode ser encontrado em <http://people.scs.carleton.ca/~dehne/projects/Cgmlib/>

Uma máquina virtual será disponibilizada como forma de auxílio à implantação do arcabouço e do modelo de programação (cgmlib). Essa VM também conterá alguns exemplos em código-fonte e executáveis. Demais informações sobre essa VM serão enviadas via Fórum do Paca.

### **2. Fase única**

Esse EP é constituído de apenas uma fase. O desafio é implementar um algoritmo de Produto de Cadeia de Matrizes. Esse algoritmo é apresentado nos seguintes materiais:

- <http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.165.2149&rep=rep1&type=pdf>
- <http://grenoble.ime.usp.br/~gold/cursos/2012/mac5742/LeonardoUFMS.pdf>

Defina o formato de entrada e de saída. Lembre-se de incluir os detalhes de entrada e de saída no relatório.

O trabalho pode ser individual ou em grupo de duas pessoas. A entrega deve ser feita pelo Paca, por meio de um arquivo ZIP ou TAR.GZ com o nome dos integrantes. Exemplos:

- MarianaAlfredo.zip
- MarianaBravoAlfredoGoldman.tar.gz

Nada de iniciais, por favor! Esse arquivo deve conter o código-fonte do programa desenvolvido. Esse mesmo código será compilado na VM. Portanto, use a VM para se

certificar do sucesso da implementação. Também é importante que seja enviado um relatório explicando detalhes sobre a utilização e a implementação do algoritmo.

### **3. Critérios de correção**

Deverão ser entregues o código-fonte e o relatório da fase única. O relatório deve conter os detalhes da implementação, os pontos positivos (facilidades) e negativos (dificuldades) encontrados durante o desenvolvimento da aplicação.