

UNIVERSIDADE FEDERAL DA BAHIA
INSTITUTO DE MATEMÁTICA E ESTATÍSTICA
DEPARTAMENTO DE CIÊNCIA DA COMPUTAÇÃO

DISCIPLINA: MATA48 – Arquitetura de Computadores
PROFESSOR: Marcos Ennes Barreto
SEMESTRE: 2019.1

INSTRUÇÕES PARA O TRABALHO PRÁTICO III

Objetivo: Construção de um programa que execute de forma paralela em ambientes multicore ou manycore, resolvendo um problema típico da álgebra linear: multiplicação matrizvetor na forma $Ax = b$.

$$Ax = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \dots & a_{mn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix} = \begin{bmatrix} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n}x_n \\ a_{21}x_1 + a_{22}x_2 + \dots + a_{2n}x_n \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn}x_n \end{bmatrix}.$$

Cada linha da **matriz A** (de tamanho $m \times n$) é multiplicada pelos elementos do **vetor x** (de tamanho n) de forma independente, como demonstrado na fórmula acima. Portanto, cada thread pode atuar multiplicando uma (ou um subconjunto de) linha(s) da matrix A pelo vetor x. O resultado é armazenado no vetor b (de tamanho m).

Matrizes esparsas CSR:

Matrizes esparsas são matrizes em que a maioria dos seus elementos são nulos. O formato CSR (Compressed Sparse Row), por sua vez, é uma forma de armazenamento que considera que nos casos em que a matriz é esparsa, é mais eficiente armazenar apenas os seus valores não-nulos. Para isto, um vetor é utilizado para armazenar todos os valores (não-nulos) da matriz, enquanto dois outros vetores são utilizados para descrever a localização de cada valor na matriz original.

Considere a matriz esparsa 5×5 :

$$\begin{bmatrix} 1 & -1 & & -3 & \\ -2 & 5 & & & \\ & & 4 & 6 & 4 \\ -3 & & 6 & 7 & \\ & 8 & & & -5 \end{bmatrix}$$

A representação desta matriz em formato CSR são 3 vetores:

values = Contendo todos os valores não-nulos;

columns = Contendo os índices das colunas de cada valor não-nulo;

rowIndex = Contendo os índices (no vetor que armazena os valores não nulos) representando o primeiro valor não-nulo de cada linha. O último elemento de rowIndex é o tamanho do vetor *values*.

Portanto, a representação desta matriz em formato CSR são os três vetores:

	0	1	2	3	4	5	6	7	8	9	10	11	12
values =	(1	-1	-3	-2	5	4	6	4	-3	6	7	8	-5)
columns =	(0	1	3	0	1	2	3	4	0	2	3	1	4)
rowIndex =	(0	3	5	8	11	13)							

- A matriz A estará armazenada em um arquivo texto na forma:

```
5 5
0 0 1.0
0 1 -1.0
0 2 0.0
0 3 -3.0
0 4 0.0
1 0 -2.0
1 1 5.0
1 2 0.0
1 3 0.0
1 4 0.0
2 0 0.0
2 1 0.0
2 2 4.0
2 3 6.0
2 4 4.0
3 0 -3.0
3 1 0.0
3 2 6.0
3 3 7.0
3 4 0.0
4 0 0.0
4 1 8.0
4 2 0.0
4 3 0.0
4 4 -5.0
```

A primeira linha do arquivo identifica a dimensão da matriz: linhas x colunas. As demais linhas representam, o índice da linha, o índice da coluna e o valor (double), respectivamente. Esta matriz deve ser lida do arquivo e representada no programa principal no formato CSR (três vetores).

- O vetor x deve ser inicializado com valores aleatórios, dentro do programa principal (**aloque um vetor x cujo tamanho seja o número de colunas da matriz A**).
- O vetor resultado b também terá tamanho igual à quantidade de linhas da matriz A .
- O vetor resultado b será escrito em um arquivo texto em disco, para fins de conferência.
- A execução do programa deve se dar na forma “./executavel CPU small.csr” para a execução em CPU e “./executavel GPU small.csr” para a execução em GPU.
- Deve ser medido o tempo de execução total da aplicação e o tempo de execução de cada kernel isoladamente, para fins de avaliação de desempenho (tempo de execução apenas da função que realiza a multiplicação paralela).