

UNIVERSIDADE DE AVEIRO
MESTRADO INTEGRADO EM ENGENHARIA COMPUTACIONAL
Teste de Computação Paralela – Parte A
15/Julho/2020

Nome: _____ Nº mec. _____

NOTE BEM: Justifique todos os passos das suas respostas. Respostas não justificadas não serão consideradas.

I. Considere uma aplicação cuja execução paralela em 10 processadores permite um *speedup* de 6. Essa aplicação é constituída em parte por código não paralelizável e no restante por código que é completamente paralelizável.

- a) Indique qual a fração da aplicação totalmente sequencial que foi paralelizada.
- b) Calcule novamente o *speedup* máximo que é possível obter agora usando no máximo 20 processadores.

II. O código seguinte pretende calcular a soma de todos os elementos de uma matriz **m** de **NxN** elementos usando OpenMP. Este código está incorreto.

```
1. sum = 0.0;
2. #pragma omp parallel for private(i) shared(sum)
3. for (i = 0; i < N; i++)
4.     for (j = 0; j < N; j++)
5.         sum += m[i][j];
```

- a) Identifique o(s) problema(s) do código anterior na determinação do soma de todos os elementos da matriz **m**.
- b) Escreva o código OpenMP correto para a determinação de todos os elementos da matriz **m**. Explique o seu código, nomeadamente todas as utilizações de diretivas OpenMP.

III. Sabendo que os valores dos registos **mm0** e **mm1** são **0xBBB0 FF00 5678 a570** e **0x0A0B 5106 A088 7B13**, respetivamente, apresente o valor do registo **mm1** após a execução das seguintes instruções.

- a) **paddb mm0, mm1** (*Add packed doubleword integers*)
- b) **paddb mm0, mm1** (*Add packed byte integers*)

IV. Considere um programa CUDA para processar uma imagem de 512x64 pixels. Cada *thread* ficará com a responsabilidade de processar um pixel da imagem. O *kernel* deste programa é lançado através de:

```
dim3 d1(A,B);
dim3 d2(C,D);
kernelA<<<d1,d2>>>(img_in, img_out);
```

- a) Indique os valores de **A**, **B**, **C** e **D** que utilizaria para processar esta imagem e indique quais os valores de **gridDim.x**, **gridDim.y**, **blockDim.x**, **blockDim.y** correspondentes.
- b) Indique como pode cada *thread* determinar as coordenadas (**x**, **y**) do pixel que irá processar.
- c) Explique por palavras suas o que é um *warp* e qual a sua importância no mecanismo de execução de um kernel CUDA.