

Objectivos:

Com esta sessão pretende-se introduzir os fundamentos do processamento paralelo.

Exercícios:

O “standard” OpenMP permite a especificação do paralelismo explorável no dado programa através de diretivas. A diretiva mais utilizada é a indicação de que as iterações de um ciclo podem ser executadas em paralelo. Esta diretiva tem a seguinte sintaxe:

```
#pragma omp parallel for
```

Para utilizar o OpenMP deve adicionar `#include<omp.h>` e acrescentar a opção `-fopenmp` na linha de comando da compilação (não se esqueça de acrescentar `-O3` na `makefile`)

- a) Inclua a diretiva `#pragma omp parallel for private(x,y)` na rotina `convolve3x1` e observe a forma como as atividades paralelas executam as iterações do ciclo. Para tal inclua a seguinte linha entre os dois ciclos e execute o programa.

```
printf("Thread %d of %d executing line %d\n", omp_get_thread_num(),
      omp_get_num_threads(), x);
```

As primitivas `omp_get_thread_num` e `omp_get_num_threads` permitem, respetivamente, saber o número da atividade paralela (*threads*) em execução e o número total. Neste caso cada atividade será executada num dos núcleos disponíveis.

Nota: pode gerar cada uma das versões simplesmente incluindo (ou não) a opção `-fopenmp` na compilação (não se esqueça de fazer `make clean` sempre que altera a `makefile`). Recolha dados para a execução da rotina `convolve3x1`, utilizando uma imagem de 256x256.

```
./convolve AC_images/abe_natsumi256.pgm result.ppm 1
```

- b) Meça o tempo de execução da rotina `convolve3x1` e calcule o ganho obtido, nesta rotina, com a execução do ciclo em paralelo. O tempo da rotina pode ser medido com o PAPI:

```
#include<papi.h>
....
long long PAPI_start, PAPI_stop;
PAPI_start = PAPI_get_real_usec();
<< segmento de código a medir >>
PAPI_stop = PAPI_get_real_usec();
printf("Time in microseconds: %lld\n", PAPI_stop - PAPI_start);
```

- c) Compare agora o ganho obtido comparando o tempo de execução de **todo** o programa. Comente os resultados.
- d) Troque a ordem dos ciclos “for” na rotina `convolve3x1` e repita a alínea b). Comente os resultados obtidos.
- e) Inclua a diretiva `#pragma omp parallel for` também ciclo interior e comente os resultados obtidos.
- f) Acrescente a cláusula `schedule(dynamic)` no final da diretiva `#pragma omp parallel for` e repita a alínea a).