

# CAP-387 / 2016

## Sexta Lista de Exercícios

### Data de Entrega: 24/1/2017

**Obejetivo:** O objetivo desta lista de exercícios é explorar as diversas possibilidades de escalonamento de loops em programas OpenMP, avaliando os correspondentes ganhos de desempenho obtidos em execuções com diferentes números de threads e distintos modos de escalonamento no sistema Santos Dumont. Para estes exercícios, deve ser previamente carregado o módulo *gcc-5/5.3*.

- 1) Considere o programa *prog.c* e a função auxiliar *mysecond.c*, disponíveis no diretório *celso.mendes2/OMP-SCHED/* do Santos Dumont. Notar que no programa *prog.c* o volume de trabalho realizado pela função  $f(i)$  é proporcional ao valor do argumento  $i$ . O número de threads utilizados é definido internamente pelo programa, na diretiva OpenMP, e pode ser configurado através de um argumento de entrada na linha de comando da execução. Pede-se o seguinte:
  - (a) Compile este programa utilizando o *Makefile* disponível no diretório indicado. Configurando o modo de escalonamento como estático/bloco (i.e. definindo a variável ambiental `OMP_SCHEDULE="static"`), execute este programa em um nó do S.Dumont com 1,2,4,8,12,16,20,24 threads, respectivamente. Apresente os tempos de execução obtidos em cada caso, e plote o *Speedup* correspondente como função do número de threads utilizados.
  - (b) Reconfigure o modo de escalonamento para cíclico (i.e. definindo a variável ambiental `OMP_SCHEDULE="static,1"`) e execute novamente o programa com os mesmos números de threads do item anterior, mostrando os novos tempos obtidos e o respectivo gráfico do *Speedup*.
  - (c) Repita o item anterior, desta vez configurando o modo de escalonamento para dinâmico (i.e. definindo a variável ambiental `OMP_SCHEDULE="dynamic"`). Mostre os novos tempos obtidos e o novo gráfico do *Speedup*.
  - (d) Apresente um gráfico único contendo as três curvas de *Speedup* obtidas nos itens (a), (b) e (c), respectivamente.
- 2) Considere agora o programa *prog2.c* existente no mesmo diretório. Note que neste caso o volume de trabalho executado pela função  $f(i)$  tem aspecto diferente daquele do primeiro exercício. Após compilar este programa com o mesmo *Makefile* usado acima, utilize-o para execuções similares às dos exercícios 1(a), 1(b), e 1(c), apresentando também um gráfico único com as novas três curvas de *Speedup* correspondentes.