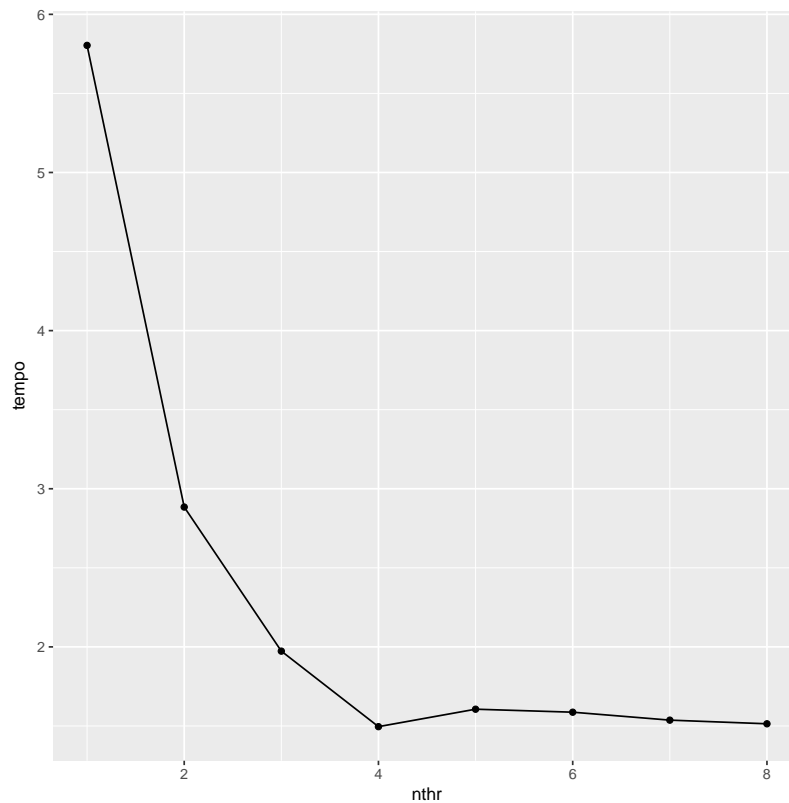


Exercícios — Pthreads — Respostas

1. Sem resposta.
2. Ver o arquivo `pthread-ex-1.c`.
3. Ver o arquivo `pthread-ex-2.c`.
4. Ver o arquivo `pthread-ex-3.c`.
5. Ver o arquivo `pthread-ex-4.c`.
6. O arquivo `pthread-ex-5.c` implementa uma solução em que N é passado como parâmetro pela linha de comando (por exemplo, para $N = 2$, deve-se executar o programa com `./pthread-ex-5 2`).

A figura abaixo mostra o tempo de execução (em segundos) para N variando entre 1 e 8, em uma máquina com 4 núcleos, considerando a média de 5 rodadas. Observa-se que o tempo cai entre 1 e 4 *threads* (número de núcleos disponíveis), e a partir desse ponto ele torna-se relativamente estável.



7. Ver o arquivo `pthread-ex-7.c`.

8. O programa pode gerar as saídas abaixo:

(a) f1: a=20	(b) f2: a=10	(c) f1: a=20	(d) f2: a=10	(e) f1: a=20	(f) f2: a=10
f2: a=10	f1: a=20	f2: a=10	f1: a=20	f2: a=10	f1: a=20
n=10	n=10	n=-10	n=-10	n=20	n=20

As saídas (a) e (b) são as mais prováveis, e envolvem apenas a mudança de ordem dos `printf()`s das linhas 13 e 20. As saídas (c) a (f) ocorrem com menor frequência, pois envolvem sequências mais improváveis de execução. Para visualizar tais sequências, primeiro é preciso considerar como as linhas 12 (`n += a`) e 19 (`n -= a`) seriam traduzidas para linguagem de máquina (`reg` é um registrador de propósito geral):

12a	<code>mov n, reg</code>	19a	<code>mov n, reg</code>
12b	<code>add a, reg</code>	19b	<code>sub a, reg</code>
12c	<code>mov reg, n</code>	19c	<code>mov reg, n</code>

Como essas instruções podem ser intercaladas, uma *thread* pode incrementar ou decrementar o valor desatualizado da variável compartilhada. Por exemplo, a saída (c) poderia ser produzida pela sequência 19a, 12, 13, 19b, 19c, 20, 31, e a saída (e) pela sequência 12a, 19, 12b, 12c, 13, 20, 31.

9. O valor final de `n` fica entre -2000 e $+2000$.