

Relatório TP-09

LPAV

Aluno: Pedro Henrique Cerávolo dos Santos
Matrícula: 21750809

O código do item 1 (“SOMATÓRIO”) já foi escrito com o relatório em mente. A implementação já gera uma matriz de tempos, de acordo com o estabelecido na especificação. Um exemplo de execução está na imagem a seguir:

```
1ms, 1ms, 0ms, 0ms, 1ms, 1ms, 0ms,  
1ms, 0ms, 1ms, 0ms, 0ms, 0ms, 0ms,  
1ms, 1ms, 1ms, 0ms, 1ms, 1ms, 1ms,  
0ms, 1ms, 1ms, 1ms, 0ms, 1ms, 1ms,  
2ms, 1ms, 2ms, 1ms, 1ms, 2ms, 2ms,  
3ms, 3ms, 2ms, 3ms, 3ms, 3ms, 3ms,  
5ms, 5ms, 6ms, 5ms, 6ms, 5ms, 5ms,  
C:\Users\PedroHC\Desktop\tp-09>
```

Formatando para uma tabela:

	1024	2048	4096	8192	16384	32768	65536
1	1 ms	1 ms	0 ms	0 ms	1 ms	1 ms	0 ms
2	1 ms	0 ms	1 ms	0 ms	0 ms	0 ms	0 ms
4	1 ms	1 ms	1 ms	0 ms	1 ms	1 ms	1 ms
8	0 ms	1 ms	1 ms	1 ms	0 ms	1 ms	1 ms
16	2 ms	1 ms	2 ms	1 ms	1 ms	2 ms	2 ms
32	3 ms	3 ms	2 ms	3 ms	3 ms	3 ms	3 ms
64	5 ms	5 ms	6 ms	5 ms	6 ms	5 ms	5 ms

É possível notar que existe um certo ponto em que adicionar mais threads desacelera o processo, já o processo de criar threads é demorado (com todo o overhead de contexto). Assim, para entradas pequenas e com poucas threads a execução é muito rápida. Porém, as threads ainda podem ajudar, caso o número delas seja equilibrado. (No exemplo, 4 threads parecem ser o número ideal.)

OBS: O código foi escrito para compilar na JDK-12.