

Universidade Presbiteriana Mackenzie
Computação Paralela – Turma 05P11
Alan Meniuk Gleizer – RA 10416804
10/05/2025

Lab 7 – Ordenação em OpenMP e OpenACC

Atividade 1 – Odd-Even Transposition Sort Paralelo

Prints execução

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 1
OpenMP: n = 10000, threads = 1 -> 0.031693 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 1
OpenMP: n = 10000, threads = 1 -> 0.032049 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 1
OpenMP: n = 10000, threads = 1 -> 0.031848 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 1
OpenMP: n = 10000, threads = 1 -> 0.031365 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 1
OpenMP: n = 10000, threads = 1 -> 0.031888 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 1
OpenMP: n = 100000, threads = 1 -> 2.964239 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 1
OpenMP: n = 100000, threads = 1 -> 2.960265 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 1
OpenMP: n = 100000, threads = 1 -> 2.958948 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 1
OpenMP: n = 100000, threads = 1 -> 2.960859 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 1
OpenMP: n = 100000, threads = 1 -> 2.953399 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 1
OpenMP: n = 500000, threads = 1 -> 70.022813 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 1
OpenMP: n = 500000, threads = 1 -> 71.211159 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 1
OpenMP: n = 500000, threads = 1 -> 72.082040 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 1
OpenMP: n = 500000, threads = 1 -> 73.168581 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 1
OpenMP: n = 500000, threads = 1 -> 71.348435 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 4
OpenMP: n = 10000, threads = 4 -> 0.020346 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 4
OpenMP: n = 10000, threads = 4 -> 0.019598 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 4
OpenMP: n = 10000, threads = 4 -> 0.019489 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 4
OpenMP: n = 10000, threads = 4 -> 0.019640 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 4
OpenMP: n = 10000, threads = 4 -> 0.019949 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 4
OpenMP: n = 100000, threads = 4 -> 0.854448 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 4
OpenMP: n = 100000, threads = 4 -> 0.855517 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 4
OpenMP: n = 100000, threads = 4 -> 0.867036 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 4
OpenMP: n = 100000, threads = 4 -> 0.854263 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 4
OpenMP: n = 100000, threads = 4 -> 0.855536 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 4
OpenMP: n = 500000, threads = 4 -> 19.319998 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 4
OpenMP: n = 500000, threads = 4 -> 19.175686 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 4
OpenMP: n = 500000, threads = 4 -> 19.162085 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 4
OpenMP: n = 500000, threads = 4 -> 19.192801 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 4
OpenMP: n = 500000, threads = 4 -> 19.201093 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 8
OpenMP: n = 10000, threads = 8 -> 0.017934 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 8
OpenMP: n = 10000, threads = 8 -> 0.017393 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 8
OpenMP: n = 10000, threads = 8 -> 0.017881 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 8
OpenMP: n = 10000, threads = 8 -> 0.017044 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 8
OpenMP: n = 10000, threads = 8 -> 0.017149 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 8
OpenMP: n = 100000, threads = 8 -> 0.522417 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 8
OpenMP: n = 100000, threads = 8 -> 0.525471 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 8
OpenMP: n = 100000, threads = 8 -> 0.522784 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 8
OpenMP: n = 100000, threads = 8 -> 0.516847 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 8
OpenMP: n = 100000, threads = 8 -> 0.520766 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 8
OpenMP: n = 500000, threads = 8 -> 10.025266 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 8
OpenMP: n = 500000, threads = 8 -> 10.020863 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 8
OpenMP: n = 500000, threads = 8 -> 10.050874 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 8
OpenMP: n = 500000, threads = 8 -> 10.012964 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 8
OpenMP: n = 500000, threads = 8 -> 10.020542 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 16
OpenMP: n = 10000, threads = 16 -> 0.017290 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 16
OpenMP: n = 10000, threads = 16 -> 0.017157 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 16
OpenMP: n = 10000, threads = 16 -> 0.016687 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 16
OpenMP: n = 10000, threads = 16 -> 0.017227 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 10000 16
OpenMP: n = 10000, threads = 16 -> 0.017345 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 16
OpenMP: n = 100000, threads = 16 -> 0.355895 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 16
OpenMP: n = 100000, threads = 16 -> 0.360680 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 16
OpenMP: n = 100000, threads = 16 -> 0.377133 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 16
OpenMP: n = 100000, threads = 16 -> 0.418397 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 100000 16
OpenMP: n = 100000, threads = 16 -> 0.359211 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 16
OpenMP: n = 500000, threads = 16 -> 5.586479 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 16
OpenMP: n = 500000, threads = 16 -> 5.705802 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 16
OpenMP: n = 500000, threads = 16 -> 5.728831 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 16
OpenMP: n = 500000, threads = 16 -> 5.703848 s
agleizer@linuxzinho:lab_OPENACC$ ./01_openMP_v1 500000 16
OpenMP: n = 500000, threads = 16 -> 5.762559 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
./01_openACC_v1 10000
OpenACC: n = 10000 -> 0.845051 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 10000
OpenACC: n = 10000 -> 0.836078 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 10000
OpenACC: n = 10000 -> 0.794672 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 10000
OpenACC: n = 10000 -> 0.786164 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 10000
OpenACC: n = 10000 -> 0.789873 s
agleizer@linuxzinho: lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
./01_openACC_v1 100000
OpenACC: n = 100000 -> 64.068529 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 100000
OpenACC: n = 100000 -> 64.119682 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 100000
OpenACC: n = 100000 -> 64.160429 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 100000
OpenACC: n = 100000 -> 64.171935 s
agleizer@linuxzinho: lab_OPENACC$ ./01_openACC_v1 100000
OpenACC: n = 100000 -> 64.175132 s
agleizer@linuxzinho: lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./01_openACC_v1 500000
```

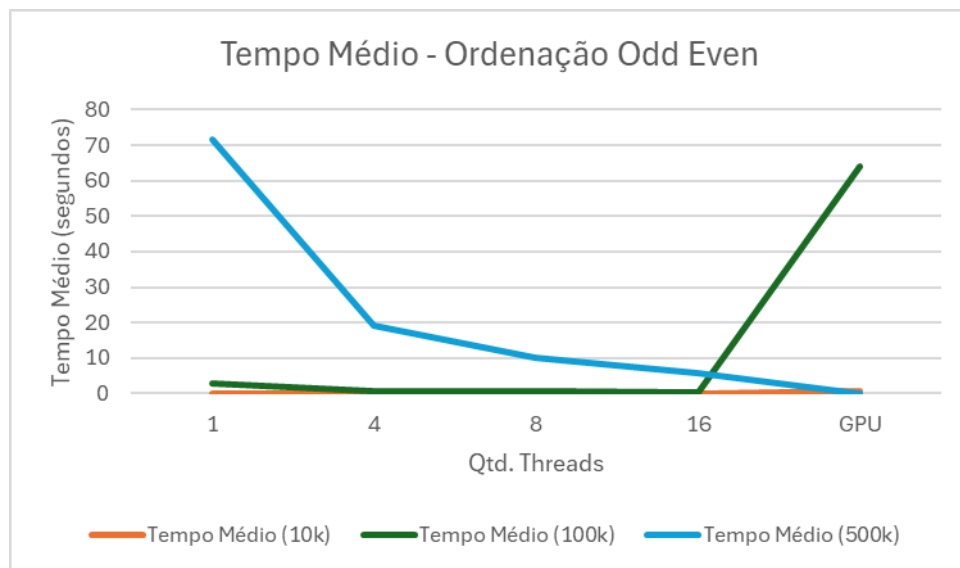
```
agleizer@linuxzinho:~$ nvidia-smi
Sat May 10 21:42:06 2025
+-----+
| NVIDIA-SMI 570.133.07                  Driver Version: 570.133.07      CUDA Version: 12.8     |
+-----+-----+-----+-----+-----+-----+
| GPU   Name                               Persistence-M | Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan   Temp   Perf           Pwr:Usage/Cap |      Memory-Usage | GPU-Util  Compute M. |
|                               |                      | MIG M. |
+-----+-----+-----+-----+-----+-----+
|  0    NVIDIA GeForce RTX 4070 ...    Off | 00000000:01:00.0  On |          N/A |
| 31%   40C    P2              60W / 285W |  860MiB / 16376MiB |    100%    Default |
|                               |                      |          N/A |
+-----+-----+-----+-----+-----+-----+

+-----+
| Processes:                               |
|  GPU   GI    CI          PID    Type   Process name                      GPU Memory |
|          ID    ID              |                 |           Usage |
+-----+-----+-----+-----+-----+-----+
|  0     N/A   N/A         3891     G   /usr/bin/gnome-shell              315MiB |
|  0     N/A   N/A         6936     G   /usr/bin/Xwayland                  4MiB |
|  0     N/A   N/A         7033     G   ...ess --variations-seed-version   78MiB |
|  0     N/A   N/A        26309     G   /usr/bin/nautilus                  44MiB |
|  0     N/A   N/A        30340     G   /usr/bin/gnome-text-editor         38MiB |
|  0     N/A   N/A        33303     G   /usr/bin/gnome-system-monitor      34MiB |
|  0     N/A   N/A        51796     C   ./01_openACC_v1                    214MiB |
+-----+-----+-----+-----+-----+-----+
agleizer@linuxzinho:~$
```

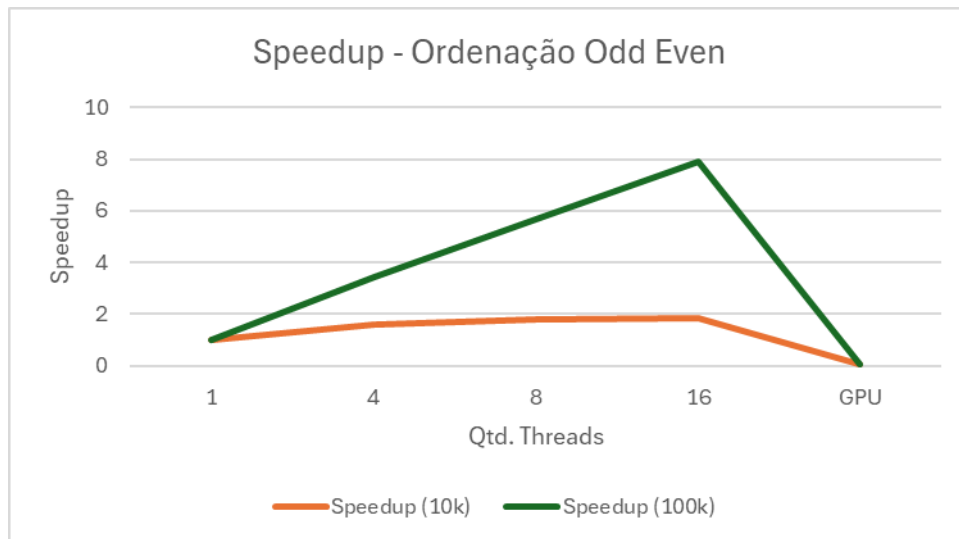

Análise dos Resultados

Para entender a efetividade das abordagens, foi necessário tabular os resultados obtidos, calcular o tempo médio para cada variação e o respectivo speedup. Os resultados podem ser observados abaixo:

Threads	Tempo Médio (10k)	Tempo Médio (100k)	Tempo Médio (500k)
1	0,0317686	2,959542	71,5666056
4	0,0198044	0,85736	19,2103326
8	0,0174802	0,521657	10,0261018
16	0,0171412	0,3742632	5,6975038
GPU	0,8103676	64,1391414	0



Threads	Speedup (10k)	Speedup (100k)	Speedup (500k)
1	1	1	1
4	1,604118277	3,451924512	3,725422516
8	1,817404835	5,67334858	7,138029019
16	1,85334749	7,907648949	12,5610457
GPU	0,039202703	0,046142526	0



O gráfico de speedup da Atividade 1 mostra um crescimento claro na performance à medida que o número de threads aumenta na versão OpenMP. Para vetores maiores ($n = 500000$), o speedup atinge valores expressivos com 16 threads, indicando que o algoritmo consegue aproveitar bem o paralelismo da CPU. No entanto, conforme aumentamos o número de threads, os ganhos marginais diminuem, o que é esperado devido ao aumento da sobrecarga de sincronização e ao fato de que Odd-Even Sort é sequencial em muitas de suas fases.

Já a versão com OpenACC apresenta desempenho drasticamente inferior. O gráfico de speedup colapsa: mesmo para vetores pequenos, o tempo de execução é dezenas de vezes maior que na versão OpenMP. Para $n = 100000$, o tempo excede 60 segundos. Isso acontece porque o algoritmo realiza trocas entre elementos vizinhos, o que gera dependências entre iterações e impede que o compilador paralelize os loops na GPU. Além disso, o custo de copiar dados entre host e device em cada fase contribui negativamente para o desempenho. Vale notar que o programa sequer apresentou resultados para $n = 500000$. Após 15 minutos, foi necessário interromper a execução.

Atividade 2 – Sample Sort Sort Paralelo

Prints execução

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 1
OpenMP: n = 10000, threads = 1 -> 0.002111 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 1
OpenMP: n = 10000, threads = 1 -> 0.002111 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 1
OpenMP: n = 10000, threads = 1 -> 0.001721 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 1
OpenMP: n = 10000, threads = 1 -> 0.001902 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 1
OpenMP: n = 10000, threads = 1 -> 0.001646 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 1
OpenMP: n = 100000, threads = 1 -> 0.009643 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 1
OpenMP: n = 100000, threads = 1 -> 0.009254 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 1
OpenMP: n = 100000, threads = 1 -> 0.009558 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 1
OpenMP: n = 100000, threads = 1 -> 0.009596 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 1
OpenMP: n = 100000, threads = 1 -> 0.009616 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 1
OpenMP: n = 500000, threads = 1 -> 0.049857 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 1
OpenMP: n = 500000, threads = 1 -> 0.046672 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 1
OpenMP: n = 500000, threads = 1 -> 0.047393 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 1
OpenMP: n = 500000, threads = 1 -> 0.047879 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 1
OpenMP: n = 500000, threads = 1 -> 0.046873 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 4
OpenMP: n = 10000, threads = 4 -> 0.002419 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 4
OpenMP: n = 10000, threads = 4 -> 0.001640 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 4
OpenMP: n = 10000, threads = 4 -> 0.002450 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 4
OpenMP: n = 10000, threads = 4 -> 0.002125 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 4
OpenMP: n = 10000, threads = 4 -> 0.001844 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 4
OpenMP: n = 100000, threads = 4 -> 0.009490 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 4
OpenMP: n = 100000, threads = 4 -> 0.010196 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 4
OpenMP: n = 100000, threads = 4 -> 0.011574 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 4
OpenMP: n = 100000, threads = 4 -> 0.008599 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 4
OpenMP: n = 100000, threads = 4 -> 0.011527 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 4
OpenMP: n = 500000, threads = 4 -> 0.043394 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 4
OpenMP: n = 500000, threads = 4 -> 0.033600 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 4
OpenMP: n = 500000, threads = 4 -> 0.046010 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 4
OpenMP: n = 500000, threads = 4 -> 0.046697 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 4
OpenMP: n = 500000, threads = 4 -> 0.041462 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 8
OpenMP: n = 10000, threads = 8 -> 0.001806 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 8
OpenMP: n = 10000, threads = 8 -> 0.001788 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 8
OpenMP: n = 10000, threads = 8 -> 0.001717 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 8
OpenMP: n = 10000, threads = 8 -> 0.002115 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 8
OpenMP: n = 10000, threads = 8 -> 0.001790 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 8
OpenMP: n = 100000, threads = 8 -> 0.008154 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 8
OpenMP: n = 100000, threads = 8 -> 0.008308 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 8
OpenMP: n = 100000, threads = 8 -> 0.007619 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 8
OpenMP: n = 100000, threads = 8 -> 0.009761 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 100000 8
OpenMP: n = 100000, threads = 8 -> 0.009427 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 8
OpenMP: n = 500000, threads = 8 -> 0.040439 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 8
OpenMP: n = 500000, threads = 8 -> 0.040392 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 8
OpenMP: n = 500000, threads = 8 -> 0.039995 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 8
OpenMP: n = 500000, threads = 8 -> 0.040111 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 500000 8
OpenMP: n = 500000, threads = 8 -> 0.040717 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 16
OpenMP: n = 10000, threads = 16 -> 0.001938 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 16
OpenMP: n = 10000, threads = 16 -> 0.002137 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 16
OpenMP: n = 10000, threads = 16 -> 0.001992 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 16
OpenMP: n = 10000, threads = 16 -> 0.001845 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openMP_v2 10000 16
OpenMP: n = 10000, threads = 16 -> 0.001741 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
./02_openMP_v2 100000 16
OpenMP: n = 100000, threads = 16 -> 0.008928 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 100000 16
OpenMP: n = 100000, threads = 16 -> 0.008613 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 100000 16
OpenMP: n = 100000, threads = 16 -> 0.008781 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 100000 16
OpenMP: n = 100000, threads = 16 -> 0.008521 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 100000 16
OpenMP: n = 100000, threads = 16 -> 0.008556 s
agleizer@linuxzinho: lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
./02_openMP_v2 500000 16
OpenMP: n = 500000, threads = 16 -> 0.040230 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 500000 16
OpenMP: n = 500000, threads = 16 -> 0.038766 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 500000 16
OpenMP: n = 500000, threads = 16 -> 0.038974 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 500000 16
OpenMP: n = 500000, threads = 16 -> 0.039100 s
agleizer@linuxzinho: lab_OPENACC$ ./02_openMP_v2 500000 16
OpenMP: n = 500000, threads = 16 -> 0.038751 s
agleizer@linuxzinho: lab_OPENACC$
```



```
agleizer@linuxzinho: lab_OPENACC
./02_openACC_v2 10000 4
OpenACC: n =10000 -> 0.157971 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 10000 4
OpenACC: n =10000 -> 0.118462 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 10000 4
OpenACC: n =10000 -> 0.113660 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 10000 4
OpenACC: n =10000 -> 0.114717 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 10000 4
OpenACC: n =10000 -> 0.112764 s
agleizer@linuxzinho:lab_OPENACC$
```

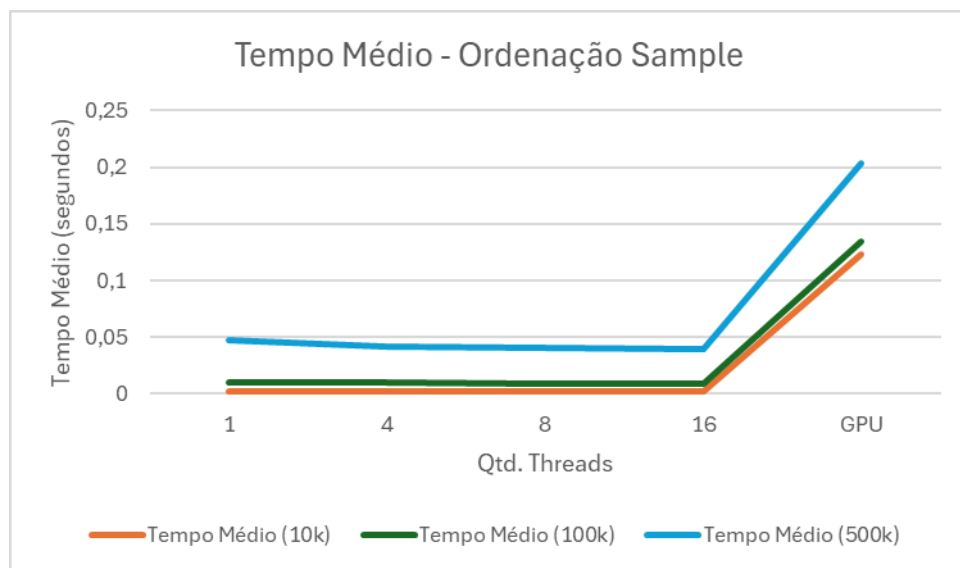
```
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 100000 4
OpenACC: n =100000 -> 0.166509 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 100000 4
OpenACC: n =100000 -> 0.128837 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 100000 4
OpenACC: n =100000 -> 0.129457 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 100000 4
OpenACC: n =100000 -> 0.126890 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 100000 4
OpenACC: n =100000 -> 0.123135 s
agleizer@linuxzinho:lab_OPENACC$
```

```
agleizer@linuxzinho: lab_OPENACC
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 500000 4
OpenACC: n =500000 -> 0.247221 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 500000 4
OpenACC: n =500000 -> 0.191729 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 500000 4
OpenACC: n =500000 -> 0.190388 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 500000 4
OpenACC: n =500000 -> 0.191854 s
agleizer@linuxzinho:lab_OPENACC$ ./02_openACC_v2 500000 4
OpenACC: n =500000 -> 0.194259 s
agleizer@linuxzinho:lab_OPENACC$
```

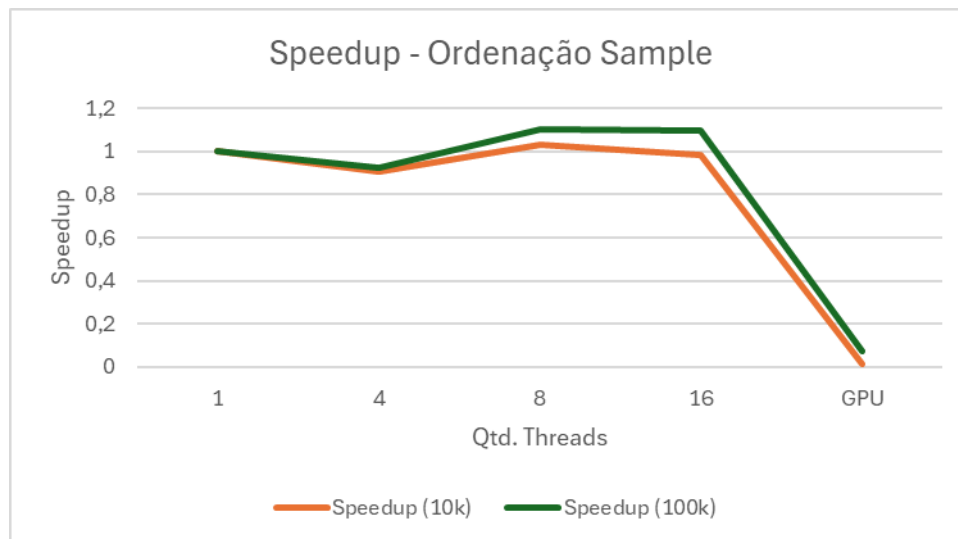
Análise dos Resultados

Para entender a efetividade das abordagens, foi necessário tabular os resultados obtidos, calcular o tempo médio para cada variação e o respectivo speedup. Os resultados podem ser observados abaixo:

Threads	Tempo Médio (10k)	Tempo Médio (100k)	Tempo Médio (500k)
1	0,0018982	0,0095334	0,0477348
4	0,0020956	0,0102772	0,0422326
8	0,0018432	0,0086538	0,0403308
16	0,0019306	0,0086798	0,0391642
GPU	0,1235148	0,1349656	0,2030902



Threads	Speedup (10k)	Speedup (100k)	Speedup (500k)
1	1	1	1
4	0,905802634	0,927626202	1,130283241
8	1,02983941	1,101643209	1,183581779
16	0,983217653	1,09834328	1,218837612
GPU	0,015368199	0,070635777	0,23504236



O gráfico de speedup da Atividade 2 revela uma dinâmica diferente. A versão serial de Sample Sort já é rápida, especialmente para tamanhos de vetor menores, o que limita os ganhos potenciais com paralelismo. Nas versões com OpenMP, mesmo usando 16 threads, o speedup é modesto. Isso sugere que os trechos sequenciais do código dominam o tempo total, tornando o ganho com múltiplas threads pequeno em relação ao overhead do paralelismo.

A versão com OpenACC mostra tempos de execução significativamente maiores que a versão CPU, mesmo na versão paralela. O gráfico mostra que, para todos os tamanhos de entrada testados, o speedup da GPU em relação à versão serial da CPU é inferior a 1, ou seja, não há speedup. Isso provavelmente ocorre porque a GPU é utilizada apenas para ordenar os baldes no final, e o custo de transferir dados para a GPU acaba não se justificando. Com isso, mesmo sendo um algoritmo paralelizável em teoria, a forma como a divisão de tarefas foi feita limita os benefícios do uso da GPU. Vale notar que o algoritmo é complexo e certamente existem implementações mais eficientes, mas além das habilidades atuais.