

HPC Homework 4
Name: Zelin Gong (zg2104)
Github Repo: https://github.com/11610309GZL/HPC_ZelinGong

Q1.

Result on Nvidia GTX 1050Ti with 4GB Graphic memory

```
(base) zelin@MAGI:~/HPC_HW4/src$ ./Mat_Vec
N: 100000000
CPU Bandwidth = 24.689165 GB/s
GPU Bandwidth = 9.698784 GB/s
Error = 256.000000
CPU Result: 100000000.000000
GPU Result: 99999744.000000
Mat: 10000 * 10000
CPU Bandwidth = 15.162248 GB/s
GPU Bandwidth = 4.332445 GB/s
Error = 117.106136
```

Result on different GPUs:

GPU	Bandwidth [GB/s]
cuda1	10.53
cuda2	11.04
cuda3	12.07
cuda4	Segmentation fault
cuda5	8.89
My GPU(GTX 1050Ti)	9.69

```
[zg2104@cuda1 HPC_Hw4]$ ./Mat_Vec
N: 100000000
CPU Bandwidth = 31.058275 GB/s
GPU Bandwidth = 10.535533 GB/s
Error = 256.000000
CPU Result: 100000000.000000
GPU Result: 99999744.000000
Mat: 10000 * 10000
CPU Bandwidth = 5.369973 GB/s
GPU Bandwidth = 3.238385 GB/s
Error = 117.105948
```

```
[zg2104@cuda2 HPC_Hw4]$ ./Mat_Vec
N: 100000000
CPU Bandwidth = 30.401288 GB/s
GPU Bandwidth = 11.038761 GB/s
Error = 256.000000
CPU Result: 100000000.000000
GPU Result: 99999744.000000
Mat: 10000 * 10000
CPU Bandwidth = 4.974220 GB/s
GPU Bandwidth = 3.460958 GB/s
Error = 117.106137
```

```
[zg2104@cuda3 HPC_Hw4]$ ./Mat_Vec
N: 100000000
CPU Bandwidth = 20.265564 GB/s
GPU Bandwidth = 12.076508 GB/s
Error = 256.000000
CPU Result: 100000000.000000
GPU Result: 99999744.000000
Mat: 10000 * 10000
CPU Bandwidth = 2.985862 GB/s
GPU Bandwidth = 4.706421 GB/s
Error = 117.106137
```

```
[zg2104@cuda4 HPC_Hw4]$ ./Mat_Vec
N: 100000000
Segmentation fault (core dumped)
[zg2104@cuda4 HPC_Hw4]$ ./Mat_Vec
N: 100000000
Segmentation fault (core dumped)
```

```
[zg2104@cuda5 HPC_Hw4]$ ./Mat_Vec
N: 100000000
CPU Bandwidth = 23.684868 GB/s
GPU Bandwidth = 8.896789 GB/s
Error = 256.000000
CPU Result: 100000000.000000
GPU Result: 99999744.000000
Mat: 10000 * 10000
CPU Bandwidth = 6.607581 GB/s
GPU Bandwidth = 3.157449 GB/s
Error = 117.105017
```

Q2.

```
(base) zelin@MAGI:~/HPC_HW4/src$ ./Jacobi
Total Time: 0.694033
Total Time GPU: 0.233461
Error = 0.002624
```

Q3.

We have made substantial progress by completing a SIMD accelerated matrix library, including all the common matrix and vector operations used in the software rasterization process, and we performed preliminary test by substituting original matrix library for the SIMD one. Moreover, we further familiarized ourselves with SSE intrinsics by building a transform matrix inversion function, as well as a general 4x4 matrix inversion using special blocking that can be better optimized for SIMD operations.