

一、实验环境：

1、操作系统内核版本：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ uname -a
Linux 22X 5.15.153.1-microsoft-standard-WSL2 #1 SMP Fri Mar 29 23:14:13 UTC 2024 x86_64 x86_64 x86_64 GNU/Linux
```

2、查看发行版本：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ lsb_release -a
No LSB modules are available.
Distributor ID: Ubuntu
Description:    Ubuntu 22.04.3 LTS
Release:        22.04
Codename:       jammy
```

3、查看 CPU 信息：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ lscpu
Architecture:          x86_64
CPU op-mode(s):        32-bit, 64-bit
Address sizes:          39 bits physical, 48 bits virtual
Byte Order:             Little Endian
CPU(s):                 20
On-line CPU(s) list:   0-19
Vendor ID:              GenuineIntel
Model name:             13th Gen Intel(R) Core(TM) i7-13700H
CPU family:             6
Model:                 186
Thread(s) per core:     2
Core(s) per socket:     10
Socket(s):              1
Stepping:               2
BogoMIPS:               5836.79
Flags:                  fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
                        mmx fxsr sse sse2 ss ht syscall nx pdpe1gb rdtscp lm constant_tsc rep_good nopl
                        xtopology tsc_reliable nonstop_tsc cpuid pni pclmulqdq vmx ssse3 fma cx16 sse4_1
                        sse4_2 x2apic movbe popcnt tsc_deadline_timer aes xsave avx f16c rdrand hypervi
                        sor lahf_lm abm 3dnowprefetch ssbd ibrs ibpb stibp ibrs_enhanced tpr_shadow vmmi
                        ept vpid ept_ad fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid rdseed adx
                        smap clflushopt clwb sha_ni xsaveopt xsavec xgetbv1 xsaves avx_vnni umip waitpk
                        g gfni vaes vpclmulqdq rdpid movdiri movdir64b fsrm md_clear serialize flush_l1d
                        arch_capabilities

Virtualization features:
Virtualization:         VT-x
Hypervisor vendor:      Microsoft
Virtualization type:    full
Caches (sum of all):
L1d:                    480 KiB (10 instances)
L1i:                    320 KiB (10 instances)
L2:                     12.5 MiB (10 instances)
L3:                     24 MiB (1 instance)
Vulnerabilities:
Gather data sampling:   Not affected
Itlb multihit:          Not affected
L1tf:                   Not affected
Mds:                    Not affected
Meltdown:               Not affected
Mmio stale data:        Not affected
Retbleed:               Mitigation; Enhanced IBRS
Spec rstack overflow:   Not affected
Spec store bypass:      Mitigation; Speculative Store Bypass disabled via prctl and seccomp
Spectre v1:             Mitigation; usercopy/swapgs barriers and __user pointer sanitization
Spectre v2:             Mitigation; Enhanced IBRS, IBPB conditional, RSB filling, PBRSB-eIBRS SW sequenc
                        e
Srbds:                  Not affected
Tsx async abort:        Not affected
```

4、查看内存：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ free
              total        used        free      shared  buff/cache   available
Mem:           7993724       727964       6784060         3248       481700       7029272
Swap:          2097152           0         2097152
```

二、test_cblas_dgemm.c 修改为行主序后
矩阵乘法计算结果不变，但影响了性能，导致性能下降
(修改前)：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ time ls
dgemm_naive.c          test_cblas_dgemm.c
lab2-Linux环境下C语言编程与矩阵乘法.pdf test_cblas_open
lab2-实验指导书.pdf   time_dgemm
link                   time_dgemm.c
row_major.c            while.c

real    0m0.002s
user    0m0.000s
sys     0m0.001s
```

(修改后)：

```
hitsz_zzx@22X:~/hpc_practice/lab2-naive-gemm$ time ls
dgemm_naive.c          test_cblas_dgemm.c
lab2-Linux环境下C语言编程与矩阵乘法.pdf test_cblas_open
lab2-实验指导书.pdf   time_dgemm
link                   time_dgemm.c
row_major.c            while.c

real    0m0.003s
user    0m0.000s
sys     0m0.002s
```

表格如下：

	256	1024	4096	8192
cblas_dgemm duration	0.082316 s	0.173487 s	1.937175 s	12.295308 s
naive_dgemm duration	0.093254 s	10.703475 s		
cblas_dgemm gflops	0.815259 GFLOPS	24.756710 GFLOPS	141.896270 GFLOPS	178.850603 GFLOPS
naive_dgemm gflops	0.359818 GFLOPS	0.200634 GFLOPS	0.103103 GFLOPS	GFLOPS

注：naïve_dgemm 在矩阵规模巨大时运行速度太慢，时间有限，故不作记录。

实现 naive_dgemm duration 和 gflops 的代码关键部分：

```
printf("m=%d,n=%d,k=%d,alpha=%lf,beta=%lf,sizeofc=%d\n", m, n, k, alpha, beta, sizeofc);
gettimeofday(&start, NULL);
naive_dgemm(m, n, k, alpha, beta, A, B, C);
//cblas_dgemm(CblasColMajor, CblasNoTrans, CblasNoTrans, m, n, k, alpha, A, Lda, B, Ldb, beta, C, L
gettimeofday(&finish, NULL);
```