

Fast Matrix Multiplication

Time limit: 5.0s **Memory limit:** 1G

How quickly can you multiply two 256×256 matrices?

Please implement the following function:

```
double multiply_matrices(double a[256][256], double b[256][256]);
```

This function should multiply the two matrices, and return the sum of squares of all entries of the resulting matrix.

The goal is to be able to perform 2^{10} of these calls in 0.8 seconds. Points will be awarded depending on how close one gets to this benchmark. NB: This benchmark is an approximation of how long it would take the benchmark to run on DMOJ infrastructure, and may be tuned after some testing.

The format of the function is intended to make grading feasible in a short time frame, as outputting $2^{10} \cdot 256^2$ values would take a non-negligible amount of time. Submissions that attempt to compute this value directly without actually performing the multiplication, or attempt to compute the product using 32-bit precision instead of 64-bit precision, will be disqualified.

If sum of squares happens to be easy to cheese, we will change the problem to ask for sum of 4th powers. We will also accept other hashing methods on doubles that are cheap and easy to verify.

You may find the following information helpful:

```
flags      : fpu vme de pse tsc msr pae mce cx8 apic sep mtrr pge mca cmov pat pse36 clflush
mmx fxsr sse sse2 ss syscall nx pdpe1gb rdtscp lm constant_tsc arch_perfmon rep_good nopl
xtopology cpuid tsc_known_freq pni pclmulqdq ssse3 fma cx16 pcid sse4_1 sse4_2 x2apic movbe
popcnt tsc_deadline_timer aes xsave avx f16c rdrand hypervisor lahf_lm abm invpcid_single pti
fsgsbase tsc_adjust bmi1 avx2 smep bmi2 erms invpcid xsaveopt arat
```