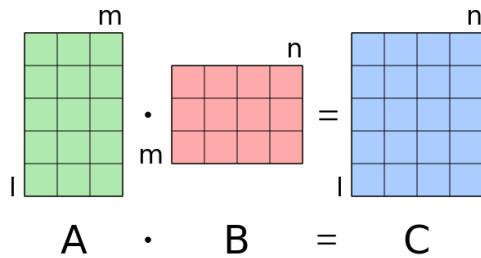


CS205 C/C++ Program Design – Project 1.

Please implement a function to compute the multiplication of two matrices. The data type of matrix elements is `float` (not `double`). As shown in the following figure¹, the resulting matrix **C**, known as the matrix product, has the number of rows of the first matrix **A** and the number of columns of the second matrix **B**.



The definition of matrix multiplication is as follows²:

If **A** is an $m \times n$ matrix and **B** is an $n \times p$ matrix,

$$\mathbf{A} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{pmatrix}, \quad \mathbf{B} = \begin{pmatrix} b_{11} & b_{12} & \cdots & b_{1p} \\ b_{21} & b_{22} & \cdots & b_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{n1} & b_{n2} & \cdots & b_{np} \end{pmatrix}$$

the *matrix product* $\mathbf{C} = \mathbf{AB}$ (denoted without multiplication signs or dots) is defined to be the $m \times p$ matrix³

$$\mathbf{C} = \begin{pmatrix} c_{11} & c_{12} & \cdots & c_{1p} \\ c_{21} & c_{22} & \cdots & c_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{m1} & c_{m2} & \cdots & c_{mp} \end{pmatrix}$$

such that

$$c_{ij} = a_{i1}b_{1j} + a_{i2}b_{2j} + \cdots + a_{in}b_{nj} = \sum_{k=1}^n a_{ik}b_{kj},$$

for $i = 1, \dots, m$ and $j = 1, \dots, p$.

Requirements:

1. (20 points) Please define a `struct` for matrices. The Struct should contain the property descriptions such as the number of rows, the number of columns, data, etc.
2. (30 points) Please implement a function which can compute the matrix product of two matrices. If the sizes of the two input matrices do not match, the function should report errors. The function must be robust and cannot crash when you input something

¹ https://en.wikipedia.org/wiki/File:Matrix_multiplication_qtll.svg

² https://en.wikipedia.org/wiki/Matrix_multiplication

incorrectly.

3. (10 points) Please measure how many milliseconds (or seconds) for the multiplication when the two matrices contain about 200M elements. **Hint:** You can follow the example at https://en.cppreference.com/w/cpp/chrono/time_point to measure the time.
4. (25 points) Improve the efficiency of your source code. Please report your methods and the improvements. **Deep analysis and comparisons are expected in the report.**
5. (10 points) Compare your implementation with OpenBLAS
<https://github.com/xianyi/OpenBLAS> Please report the differences on results and speed.
Hint: function `cblas_sgemm()` in OpenBLAS.
6. (5 points) Please host your source code at GitHub.com. you can just put a link in the report. If you do not host your source code at GitHub.com, please upload your source with your report to Blackboard.
7. Your total score will also be affected by your source code quality and report quality.

Rules:

1. Please submit your assignment report before its deadline. After the deadline (even 1 second), **0 score!**
2. Do not code your program unnecessarily complex. **Simple is beautiful!**
3. Please pay more attention to your **code style**. After all this is not ACM-ICPC contest. You have enough time to write code with both correct result and good code style. You will get deduction if your code style is terrible. You can read Google C++ Style Guide (<http://google.github.io/styleguide/cppguide.html>) or some other guide for code style.