Matrix Multiplication

We will start with the following assumptions:

- The matrices are square and of the same dimension
- The matrices will be read in, and stored, as a one-dimensional array
- Index starts at $k = 0$

As an example, if we consider two 4×4 matrices A and B, each having 16 elements, then the resulting product will be 4×4 as well. If we break this down in terms of elements of the matrices, we have the identity

$$\begin{bmatrix} A_0 & A_1 & A_2 & A_3 \\ A_4 & A_5 & A_6 & A_7 \\ A_8 & A_9 & A_{10} & A_{11} \\ A_{12} & A_{13} & A_{14} & A_{15} \end{bmatrix} \cdot \begin{bmatrix} B_0 & B_1 & B_2 & B_3 \\ B_4 & B_5 & B_6 & B_7 \\ B_8 & B_9 & B_{10} & B_{11} \\ B_{12} & B_{13} & B_{14} & B_{15} \end{bmatrix} = \begin{bmatrix} C_0 & C_1 & C_2 & C_3 \\ C_4 & C_5 & C_6 & C_7 \\ C_8 & C_9 & C_{10} & C_{11} \\ C_{12} & C_{13} & C_{14} & C_{15} \end{bmatrix} \tag{1}$$

From linear algebra, we know that the standard formula for matrix multiplication is given by

$$C_{i,j} = \sum_{k=1}^{n} A_{i,k}\, B_{k,j} \tag{2}$$

where the i and j represent the row and column, respectively, of the matrices in question. The goal here, is to combine equations (1) and (2) so that C code can be written to perform matrix multiplication without the use of two-dimensional arrays. Thus, the indices i and j will be rewritten in terms of only one index, let us call it k. The k index will range from 0 to $n-1$, where n is the number of elements in each array. For the 4×4 example, we have $n = 16$ and thus $k \in \{0, 1, \ldots, 15\}$.

So remember, to get the entry in row i column j of the matrix C, we multiply entry-wise, row i of A by column j of B. So we need a formula that relates the row and column of a matrix to the entry k in the array. First, we will define N, which gets used quite a bit in future work:

$$N = \sqrt{n} \tag{3}$$

Note that there are N rows, where the first row has entries $0, 1, \ldots, N-1$, the second row has entries $N, N+1, \ldots, 2N-1$ etc... Thus, the row number r is given by

$$r = \left\lfloor \frac{k}{N} \right\rfloor + 1 = \text{floor}\left(\frac{k}{N}\right) + 1, \tag{4}$$

where the ceiling function rounds to the next largest integer. As examples

$$\left\lfloor \frac{7}{4} \right\rfloor + 1 = \lfloor 1.75 \rfloor + 1 = 1 + 1 = 2,$$
$$\left\lfloor \frac{8}{4} \right\rfloor + 1 = \lfloor 2 \rfloor + 1 = 2 + 1 = 3,$$
$$\left\lfloor \frac{9}{4} \right\rfloor + 1 = \lfloor 2.25 \rfloor + 1 = 2 + 1 = 3$$

Notice that if you look at entries 7, 8 and 9 in each matrix in (1), we do get the correct row number per entry.

We now move on to the columns, which means we must deal with remainders after division. For example, if we look at entries 7, 8 and 9 again, we have

$$7 = 4 \cdot 1 \text{ r } 3,$$
$$8 = 4 \cdot 2 \text{ r } 0,$$
$$9 = 4 \cdot 2 \text{ r } 1$$

notice that the remainder is always one less than the column in question. Thus, we can define the column number c for each entry as

$$c = (k\%N) + 1 = \text{fmod}(k, N) + 1 \tag{5}$$

So now we have the row r, and column c, corresponding to entry k in the matrix C. So we next need to multiply the entries in row r of A by the entries in column c of B. This begs the next question — What are the indices of the entries in row r of A and column c of B? Notice that the rows of A have N entries:

$$
\begin{aligned}
&\text{row } 1 : \{0, 1, \ldots, N-1\} \\
&\text{row } 2 : \{N, N+1, \ldots, 2N-1\} = \{0, 1, \ldots, N-1\} + 1 \cdot N \\
&\text{row } 3 : \{2N, 2N+1, \ldots, 3N-1\} = \{0, 1, \ldots, N-1\} + 2 \cdot N \\
&\qquad \vdots \\
&\text{row } r : \{(r-1)N, (r-1)N+1, \ldots, rN-1\} = \{0, 1, \ldots, N-1\} + (r-1) \cdot N
\end{aligned}
\tag{6}
$$

Similarly, for the columns of B, we have

$$
\begin{aligned}
&\text{column } 1 : \{0, N, 2N, \ldots, (N-1)N\} \\
&\text{column } 2 : \{1, N+1, 2N+1, \ldots, (N-1)N+1\} = \{0, N, 2N, \ldots, (N-1)N\} + 1 \\
&\text{column } 3 : \{2, N+2, 2N+2, \ldots, (N-1)N+2\} = \{0, N, 2N, \ldots, (N-1)N\} + 2 \\
&\qquad \vdots \\
&\text{column } r : \{(c-1), N+(c-1), 2N+(c-1), \ldots, (N-1)N+(c-1)\} = \{0, N, 2N, \ldots, (N-1)N\} + (c-1)
\end{aligned}
\tag{7}
$$

Note that in the definition of the entries in row r and column c everything can be summed in terms of N. Thus, we have

$$
C_k = \sum_{s=0}^{N-1} A[s + (r-1)N] \cdot B[sN + (c-1)]
\tag{8}
$$

Note that N, r and c are already predetermined and are functions of k, and thus this sum is completely determined to be a function of the entry value k.

As an example, we will attempt to find a formula for C_{11} in (1). In this case,

$$
N = \sqrt{16} = 4, \;\; r = \left\lfloor \frac{11}{4} \right\rfloor + 1 = 2 + 1 = 3, \;\; c = (11\%4) + 1 = 3 + 1 = 4
$$

and (8) becomes

$$
\begin{aligned}
C_{11} &= \sum_{s=0}^{3} A[s+8] \cdot B[4s+3] \\
&= A[8]\, B[3] + A[9]\, B[7] + A[10]\, B[11] + A[11]\, B[15]
\end{aligned}
$$

If we look at the original matrix equation once again, but this time bolding row 3 of A, column 4 of B and entry 11 of C, we notice right away that we have indeed arrived at the correct formula!

$$
\begin{bmatrix} A_0 & A_1 & A_2 & A_3 \\ A_4 & A_5 & A_6 & A_7 \\ \mathbf{A_8} & \mathbf{A_9} & \mathbf{A_{10}} & \mathbf{A_{11}} \\ A_{12} & A_{13} & A_{14} & A_{15} \end{bmatrix} \cdot \begin{bmatrix} B_0 & B_1 & B_2 & \mathbf{B_3} \\ B_4 & B_5 & B_6 & \mathbf{B_7} \\ B_8 & B_9 & B_{10} & \mathbf{B_{11}} \\ B_{12} & B_{13} & B_{14} & \mathbf{B_{15}} \end{bmatrix} = \begin{bmatrix} C_0 & C_1 & C_2 & C_3 \\ C_4 & C_5 & C_6 & C_7 \\ C_8 & C_9 & C_{10} & \mathbf{C_{11}} \\ C_{12} & C_{13} & C_{14} & C_{15} \end{bmatrix}
$$