Matrix Multiplication Guide

What is Matrix Multiplication?

A fundamental operation in linear algebra where two matrices are multiplied to produce a new matrix.

How Matrix Multiplication Works

Given two matrices:

- A (size: m x n)
- B (size: n x p)

The resulting matrix C = A x B has dimensions m x p.

Each element C[i][j] is computed as the sum of products of the i-th row of A and the j-th column of B:
C[i][j] = sum(A[i][k] * B[k][j]) for k from 1 to n

Example Calculation

Let's multiply two 2x2 matrices:

A = [[1, 2], B = [[5, 6],
 [3, 4]] [7, 8]]

Result:
C[0][0] = 1*5 + 2*7 = 19
C[0][1] = 1*6 + 2*8 = 22
C[1][0] = 3*5 + 4*7 = 43
C[1][1] = 3*6 + 4*8 = 50

Final matrix:
[[19, 22],
 [43, 50]]

Applications in Computing

- Deep Learning: Neural network operations
- Computer Graphics: 3D transformations
- Scientific Computing: Solving linear systems
- Data Science: PCA, recommendation systems

Python Implementations

NumPy (CPU):

```
>>> import numpy as np
>>> A = np.random.rand(1000, 1000)
>>> B = np.random.rand(1000, 1000)
>>> C = A @ B  # Matrix multiplication
```

PyTorch (GPU):

```
>>> import torch
>>> A = torch.rand(1000, 1000).cuda()
>>> B = torch.rand(1000, 1000).cuda()
>>> C = torch.mm(A, B)  # GPU-accelerated
```