

# EXPERIMENT 1

## Fundamentals of Tensor Operations using PyTorch and NumPy

---

### OBJECTIVE

To understand and implement basic tensor operations using PyTorch and NumPy including creation, manipulation, arithmetic, reshaping, broadcasting, and memory-efficient in-place operations.

---

### THEORY

PyTorch and NumPy are essential libraries for numerical and tensor computations in Python. PyTorch provides GPU-accelerated operations and automatic differentiation, making it suitable for deep learning. NumPy offers highly optimized operations on n-dimensional arrays and is widely used in scientific computing.

**Tensors** are generalizations of scalars, vectors, and matrices. PyTorch's `Tensor` and NumPy's `ndarray` share similar behavior, but PyTorch tensors can utilize GPU acceleration.

Common tensor operations include:

- Creation of 1D, 2D, 3D tensors
- Element-wise arithmetic
- Dot products and matrix multiplications
- Indexing and slicing
- Reshaping with `.view()`, `.reshape()`, `.unsqueeze()`, `.squeeze()`
- Broadcasting
- In-place and out-of-place operations