ES336: Semester II [2025-20226] Assignment 01
Due Date: 25-01-2026 (11:59pm)   0 points

Objective: Learn how the choice of programming language and compiler affects the execution time of a program. Concepts of compilation stages, optimizations and cross-compilation.
Note: Make appropriate assumptions when required. Assumptions should not contradict what has been given in the problem statement.

---

1. Write a C program to multiply square matrices of size **N x N** where N is at-least **1024.** Use either int or float data types. Compute the time needed to execute multiplication. You may use and refer *timespec* or *timeval* functions. As a part of compilation generate the intermediate. Observe and highlight the changes in .i and .s files. Execute the binary program with '*time*'' command.

```
for (i = 0; i < N; i++)
  for (j = 0; j < N; j++)
    for (k = 0; k < N; k++)
      C[i][j] += A[i][k] * B[k][j];
```

2. Similarly write your own Python program using nested loops for matrix multiplication of same size **N x N**. Use timeit module to compute the execution time of the multiplication. Execute the python code with '*time*'' command.

```
for i in range(N):
  for j in range(N):
    for k in range(N):
      C[i][j] += A[i][k] * B[k][j]
```

3. Now, compare the execution time of the C & Python programs. Reason the outputs of time command, and the execution time observed by you in each of the case for multiplication.

4. You may now use the NumPy's built-in matrix multiplication function and measure the execution time. Compare the results and reason your findings.

5. We will now use compiler optimization flags, say using gcc, we can recompile the C code with different optimization levels (-O2 and -O3) and measure the execution time. You might also want to look at the intermediate assembly code (.s file) and compare the differences.

```
#gcc -O2 matmul.c -o matmul_O2.out
```

Bonus: Learn to cross-compile your c-code for MIPS architecture. ii) Try to estimate the Instruction Count and CPI for each of the Q1, Q2, Q4 and Q5 approaches.