# Bahria University,

## Karachi Campus



LAB EXPERIMENT NO.

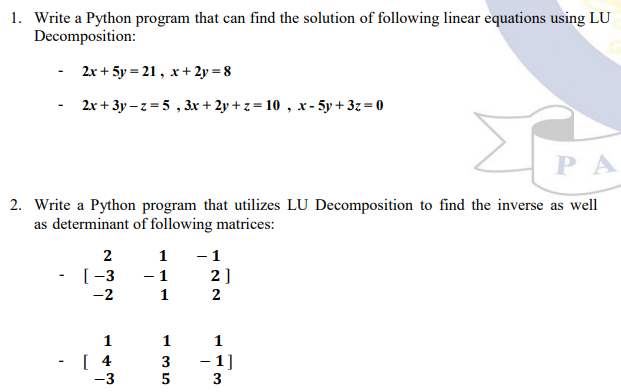
**\_13\_**

LIST OF TASKS

|  |  |
| --- | --- |
| **TASK NO** | **OBJECTIVE** |
| 01 |  |
| 02 | A math problem with numbers and equations  Description automatically generated with medium confidence |

Submitted On:

Date: 24/12/2024

**Task No 01:** 

**Solution:**

import numpy as np

from scipy.linalg import lu, lu\_factor, lu\_solve

def solve\_linear\_system(A, b):

    lu, piv = lu\_factor(A)

    x = lu\_solve((lu, piv), b)

    return x

A1 = np.array([[2, 5], [1, 2]])

b1 = np.array([21, 8])

solution1 = solve\_linear\_system(A1, b1)

A2 = np.array([[2, 3, -1], [3, 2, 1], [1, -5, 3]])

b2 = np.array([5, 10, 0])

solution2 = solve\_linear\_system(A2, b2)

print("Solutions to linear equations:")

print(f"System 1 solution: {solution1}")

print(f"System 2 solution: {solution2}")

**Output:**

**A black text on a white background

Description automatically generated**

**Task No 02:**

A math problem with numbers and equations

Description automatically generated with medium confidence

**Solution:**

def inverse\_and\_determinant(A):

    P, L, U = lu(A)

    determinant = np.prod(np.diag(U)) \* (-1)\*\*np.sum(P != np.eye(P.shape[0]))

    inverse = np.linalg.inv(A)  # Using numpy for simplicity

    return inverse, determinant

A3 = np.array([[2, 1, -1], [-3, -1, 2], [-2, 1, 2]])

inverse1, determinant1 = inverse\_and\_determinant(A3)

A4 = np.array([[1, 1, 1], [4, 3, -1], [-3, 5, 3]])

inverse2, determinant2 = inverse\_and\_determinant(A4)

print("\nMatrix inverses and determinants:")

print(f"Matrix 1 inverse:\n{inverse1}\nDeterminant: {determinant1}")

print(f"Matrix 2 inverse:\n{inverse2}\nDeterminant: {determinant2}")

**Output:**

**A screenshot of a computer

Description automatically generated**