12.339

Bhargav - EE25BTECH11013

October 2, 2025

Question:
If

$$\mathbf{A} = \begin{pmatrix} 3 & -3 \\ -3 & 4 \end{pmatrix} \tag{1}$$

then

$$\det\left(-\mathbf{A}^2 + 7\mathbf{A} - 3\mathbf{I}\right) \tag{2}$$

is

Solution

$$\det(\mathbf{A}) = 3, \mathrm{Tr}(\mathbf{A}) = 3 + 4 = 7 \tag{3}$$

(where $\mathrm{Tr}(\mathbf{A})$ represents the trace of matrix $\mathbf{A}$, i.e. the sum of the diagonal entries of $\mathbf{A}$)

Solution

The determinant of the expression can be found out by using the Cayley-Hamilton theorem.

$$\mathbf{A}^2 - (\text{Tr}(\mathbf{A}))\mathbf{A} + \det(\mathbf{A})\mathbf{I} = 0 \tag{4}$$

$$\mathbf{A}^2 - 7\mathbf{A} + 3\mathbf{I} = 0 \implies -\mathbf{A}^2 + 7\mathbf{A} - 3\mathbf{I} = 0 \tag{5}$$

$$\therefore \det\left(-\mathbf{A}^2 + 7\mathbf{A} - 3\mathbf{I}\right) = 0 \tag{6}$$

C Code

```c
#include <stdio.h>

double det(double *mat) {
    return mat[0]*mat[3] - mat[1]*mat[2];
}
```

Python + C Code

```
import numpy as np
import ctypes
lib = ctypes.CDLL("./libcode.so")
lib.det.argtypes = [ctypes.POINTER(ctypes.c_double)]
lib.det.restype = ctypes.c_double

A = np.array([[3, -3],
              [-3, 4]], dtype=np.float64)

I = np.eye(2, dtype=np.float64)
expr = -A @ A + 7*A - 3*I
mat_flat = expr.flatten()
det_value = lib.det(mat_flat.ctypes.data_as(ctypes.POINTER(ctypes
    .c_double)))
print("Matrix (-A^2 + 7A - 3I):\n", expr)
print("Determinant =", det_value)
```

Python Code

```
import numpy as np
A = np.array([[3, -3],
              [-3, 4]])
I = np.eye(2)

expr = -A @ A + 7*A - 3*I

det_value = np.linalg.det(expr)

print("Matrix A:\n", A)
print("Matrix (-A^2 + 7A - 3I):\n", expr)
print("Determinant =", det_value)
```