Problem 5.8.21

ee25btech11023-Venkata Sai

September 16, 2025

Problem

The Sum of the digits of a two-digit number is 9. Also, nine times this number is twice the number obtained by reversing the order of the digits. Find the number

Equations

Let $\mathbf{x}$ be the matrix that contains the digits of the required number N

$$N = \begin{pmatrix} 10 & 1 \end{pmatrix} \mathbf{x} \tag{2.1}$$

Given Sum of the digits of a two-digit number is 9

$$\begin{pmatrix} 1 & 1 \end{pmatrix} \mathbf{x} = 9 \tag{2.2}$$

Nine times this number is twice the number obtained by reversing the order of the digits.

$$9 \begin{pmatrix} 10 & 1 \end{pmatrix} \mathbf{x} = 2 \begin{pmatrix} 1 & 10 \end{pmatrix} \mathbf{x} \tag{2.3}$$

$$\begin{pmatrix} 90-2 & 9-20 \end{pmatrix} \mathbf{x} = 0 \tag{2.4}$$

$$\begin{pmatrix} 88 & -11 \end{pmatrix} \mathbf{x} = 0 \tag{2.5}$$

$$11 \begin{pmatrix} 8 & -1 \end{pmatrix} \mathbf{x} = 0 \implies \begin{pmatrix} 8 & -1 \end{pmatrix} \mathbf{x} = 0 \tag{2.6}$$

Augmented matrix

$$\begin{pmatrix} 1 & 1 \\ 8 & -1 \end{pmatrix} \mathbf{x} = \begin{pmatrix} 9 \\ 0 \end{pmatrix} \tag{2.7}$$

Augmented Matrix:

$$\left(\begin{array}{cc|c} 1 & 1 & 9 \\ 8 & -1 & 0 \end{array}\right) \xrightarrow{R_2 \to R_2 - 8R_1} \left(\begin{array}{cc|c} 1 & 1 & 9 \\ 0 & -9 & -72 \end{array}\right) \xrightarrow{R_1 \to R_1 + \frac{1}{9}R_2} \left(\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & -1 & -8 \end{array}\right) \tag{2.8}$$

$$\left(\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & -1 & -8 \end{array}\right) \xrightarrow{R_2 \to -R_2} \left(\begin{array}{cc|c} 1 & 0 & 1 \\ 0 & 1 & 8 \end{array}\right) \tag{2.9}$$

$$\mathbf{x} = \begin{pmatrix} 1 \\ 8 \end{pmatrix} \tag{2.10}$$

$$N = \begin{pmatrix} 10 & 1 \end{pmatrix} \begin{pmatrix} 1 \\ 8 \end{pmatrix} = 10 + 8 = 18 \tag{2.11}$$

Conclusion

Hence the Required Number is 18

Plot

C Code

```
void get_system_coeffs(double* out_data) {

    out_data[0] = 1.0;
    out_data[1] = 1.0;
    out_data[2] = 8.0;
    out_data[3] = -1.0;
    out_data[4] = 9.0;
    out_data[5] = 0.0;
}
```

Python Code for Calling

```
import ctypes
import sympy

def solve_number_problem():

    lib = ctypes.CDLL('./code.so')
    double_array_6 = ctypes.c_double * 6
    # The C function will fill an array of 6 doubles
    lib.get_system_coeffs.argtypes = [ctypes.POINTER(ctypes.
        c_double)]

    out_data_c = double_array_6()
    lib.get_system_coeffs(out_data_c)

    # Unpack the raw coefficient data from C
    m_coeffs = list(int(v) for v in out_data_c)[:4]
    c_coeffs = list(int(v) for v in out_data_c)[4:]
```

Python Code for Solving

```
 aug_M = sympy.Matrix([
        [m_coeffs[0], m_coeffs[1], c_coeffs[0]],
        [m_coeffs[2], m_coeffs[3], c_coeffs[1]]
    ])
    print("Initial Augmented Matrix:\n", aug_M)
    rref_matrix, _ = aug_M.rref()
    print("\nReduced Row Echelon Form:\n", rref_matrix)

    x_digit = rref_matrix[0, 2]
    y_digit = rref_matrix[1, 2]

    return int(x_digit), int(y_digit)
x, y = solve_number_problem()

    # Calculate the final two-digit number
number = 10 * x + y

print(f"The number is: {number}")
```