10.3.11

BEERAM MADHURI - EE25BTECH11012

October 2025

Find the normal at the point $(1, 1)$ on the curve

$$2y + x^2 = 3 \tag{1}$$

finding the Normal:

Let $F = 2y + x^2 - 3 = 0$
gradient Vector is:

$$\begin{bmatrix} \frac{\partial F}{\partial x} \\ \frac{\partial F}{\partial y} \end{bmatrix} = \begin{bmatrix} 2x \\ 2 \end{bmatrix} \tag{2}$$

Normal Vector at $(1, 1)$ is:

$$n = \begin{bmatrix} 2 \\ 2 \end{bmatrix} \tag{3}$$

let m be tangent vector,

$$\text{if } n = \begin{bmatrix} a \\ b \end{bmatrix} \tag{4}$$

$$\text{then } m = \begin{bmatrix} -b \\ a \end{bmatrix} \tag{5}$$

$$\therefore m = \begin{bmatrix} -2 \\ 2 \end{bmatrix} \tag{6}$$

let

$$p = \begin{bmatrix} x \\ y \end{bmatrix} \& p_0 = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \tag{7}$$

$$m^\top(P - P_0) = 0 \tag{8}$$

Substituting the values:-

$$\begin{bmatrix} -2 & 2 \end{bmatrix} \begin{bmatrix} x - 1 \\ y - 1 \end{bmatrix} = 0 \tag{9}$$

$$-2(x - 1) + 2(y - 1) = 0 \tag{10}$$

$$y = x \tag{11}$$

Hence equation of normal to $2y + x^2 - 3 = 0$ at $(1, 1)$ is $y = x$.

Python Code

```python
import numpy as np
import matplotlib.pyplot as plt

# Create the figure and axis
fig, ax2 = plt.subplots()
fig.suptitle('Graphs of Normal to the Curve', fontsize=16)
# Curve: 2y + x^2 = 3 => y = (3 - x^2)/2
# Point: (1, 1)
# Normal Line: y = x
# Tangent Line at (1,1): y = -x + 2
```

Python Code

```
# Generate x-values for the curve
x_curve2 = np.linspace(-3, 4, 400)
y_curve2 = (3 - x_curve2**2) / 2

# Generate x-values for the lines
x_line2 = np.linspace(-2, 4, 100)
y_normal = x_line2
y_tangent2 = -x_line2 + 2
```

Python Code

```python
# Plot the curve, normal, tangent, and the point
ax2.plot(x_curve2, y_curve2, label='Curve: $2y+x^2=3$', color='blue')
ax2.plot(x_line2, y_normal, label='Normal: $y=x$', color='green')
ax2.plot(x_line2, y_tangent2, label='Tangent: $y=-x+2$', color='red', linestyle='--')
ax2.scatter([1], [1], color='purple', s=100, zorder=5, label='Point (1, 1)')
```

Python Code

```python
# Formatting for the plot
ax2.set_title('10.3.11: Normal to $2y+xˆ2=3$', fontsize=12)
ax2.set_xlabel('$x$')
ax2.set_ylabel('$y$')
ax2.axhline(0, color='black', linewidth=0.5)
ax2.axvline(0, color='black', linewidth=0.5)
ax2.grid(True, linestyle='--', alpha=0.7)
ax2.legend()
ax2.set_aspect('equal', adjustable='box')
ax2.set_xlim([-2, 4])
ax2.set_ylim([-2, 4])
```

Python Code

```
# Display the figure
plt.tight_layout(rect=[0, 0.03, 1, 0.95])
plt.show()
```

C Code

```c
#include <stdio.h>

int main() {
    // --- Given Problem Data ---
    // The point (x1, y1) on the curve 2y + x^2 = 3.
    double x1 = 1.0;
    double y1 = 1.0;
    // --- Calculations ---
    // 1. Find the slope of the tangent from the derivative dy/dx
         = -x.
    double m_tangent = -x1;
```

C Code

```
// A horizontal tangent (slope=0) means the normal is a
    vertical line.
if (m_tangent == 0) {
    printf("The equation of the normal line is: x = %.2f\n",
        x1);
} else {
    // 2. The slope of the normal is the negative reciprocal.
    double m_normal = -1.0 / m_tangent;
```

C Code

```c
// 3. The y-intercept 'c' is calculated from y = mx + c.
double c = y1 - m_normal * x1;

// --- Print the Final Answer ---
// This logic helps format the equation cleanly (e.g.,
    printing "y = x"
// instead of "y = 1.00x + 0.00").
printf("The equation of the normal line is: ");
```

C Code

```c
    if (m_normal == 1 && c == 0) {
        printf("y = x\n");
    } else if (m_normal == -1 && c == 0) {
        printf("y = -x\n");
    } else {
        printf("y = %.2fx", m_normal);
        if (c > 0) {printf(" + %.2f\n", c);
        } else if (c < 0) {printf(" - %.2f\n", -c);
        } else {printf("\n");
        }}}
return 0;}
```

Python and C Code

```python
from ctypes import c_double

def main():
    # --- Given Problem Data ---
    x1 = c_double(1.0)
    y1 = c_double(1.0)

    # --- Calculations ---
    m_tangent = c_double(-x1.value) # dy/dx = -x
```

Python and C Code

```python
if m_tangent.value == 0:
    print(f"The equation of the normal line is: x = {x1.value
        :.2f}")
else:
    m_normal = c_double(-1.0 / m_tangent.value)
    c_intercept = c_double(y1.value - m_normal.value * x1.
        value)

    # --- Print the Final Answer ---
    print("The equation of the normal line is: ", end="")
```

Python and C Code

```python
if m_normal.value == 1 and c_intercept.value == 0:
    print("y = x")
elif m_normal.value == -1 and c_intercept.value == 0:
    print("y = -x")
else:
    print(f"y = {m_normal.value:.2f}x", end="")
    if c_intercept.value > 0:
        print(f" + {c_intercept.value:.2f}")
```

```python
            elif c_intercept.value < 0:
                print(f" - {abs(c_intercept.value):.2f}")
            else:
                print()

if __name__ == "__main__":
    main()
```
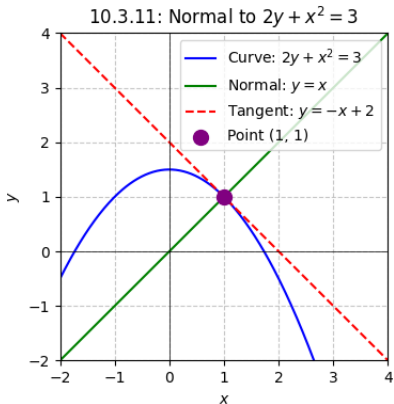
Graphs of Normal to the Curve

Figure: Plot