```
import math


# Function to calculate the forward differences
def forward_difference(y_values):
    # Create a table for forward differences
    n = len(y_values)
    difference_table = [y_values]

    # Calculate forward differences
    for i in range(1, n):
        temp = []
        for j in range(n - i):
            temp.append(difference_table[i - 1][j + 1] - difference_table[i - 1][j])
        difference_table.append(temp)

    return difference_table


# Function to apply Newton's Forward Interpolation
def newton_forward_interpolation(x_values, y_values, t):
    # Calculate the forward differences table
    difference_table = forward_difference(y_values)

    # Calculate the value of u = (t - x0) / h where h is the difference between x values
    h = x_values[1] - x_values[0]  # Assuming the x values are equally spaced
    u = (t - x_values[0]) / h
```

```python
    # Initialize the interpolated value
    interpolated_value = y_values[0]

    # Apply the Newton's Forward Formula
    u_term = u
    for i in range(1, len(x_values)):
        # Factorial term is divided by i! and added to the interpolated value
        interpolated_value += (u_term * difference_table[i][0]) / math.factorial(i)
        # Update u_term for the next iteration
        u_term *= (u - i)

    return interpolated_value

# Main function
def main():
    # Given data points (time and voltage across the capacitor)
    x_values = [0, 1, 2, 3, 4, 5]  # Time in seconds
    y_values = [0, 2.5, 4.5, 5.5, 5.8, 6.0]  # Voltage across capacitor in volts

    # Input the time at which we need to estimate the voltage
    t = float(input("Enter the time (t) to estimate the voltage (in seconds): "))

    # Apply Newton's Forward Interpolation to estimate the voltage
    estimated_voltage = newton_forward_interpolation(x_values, y_values, t)

    # Output the result
```

```python
    print(f"The estimated voltage across the capacitor at {t} seconds is: {estimated_voltage}")


# Run the main function
if __name__ == "__main__":
    main()
```