```python
# ESE 531: HW1 Problem 2
# M-point Moving Average Filter

# import libraries
from turtle import color
import numpy as np
import matplotlib.pyplot as plt

# PART A

# generate defined signal
s = np.array([2 * n * 0.9 ** n for n in range(101)])

# PART B

# random gaussain noise
w = np.array([np.random.normal(0, 1) for n in range(101)])

# PART C

# define signal x as s plus noise
x = s + w

# plot all three signals
plt.stem(s)
plt.xlabel('n')
plt.ylabel('$s[n]$')
plt.show()

plt.stem(w)
plt.xlabel('n')
plt.ylabel('$w[n]$')
plt.show()

plt.stem(x)
plt.xlabel('n')
plt.ylabel('$x[n]$')
plt.show()


# PART D

# Moving Average Function
'''
Arguments:
    x: input signal
    m: window size of moving average
Output:
    y: averaged signal
'''
def moving_avg(x, m):

    # the impulse response of the moving average filter is a scaled and shifted window
    h = np.ones(m) / m

    # to apply the filter, we simply convolve the signal with the impulse response
    y = np.convolve(x, h)

    return y

# apply 5-point moving average filter to x[n]
y = moving_avg(x, 5)
plt.stem(y, label='$y[n]$', markerfmt='ro', linefmt='r-')
plt.stem(s, label='$s[n]$', markerfmt='bo')
plt.xlabel('n')
plt.ylabel('signal value')
plt.legend()
```

```
plt.show()

# PART E

# generate interference signal
f = 0.2
w_int = np.cos([2 * np.pi * f * n for n in range(101)])

# interfered signal
x_int = s + w_int

# Filter interfered signal with moving average filter of variable window sizes
m_list = [4, 5, 6]
for m in m_list:
    y_int = moving_avg(x_int, m)
    plt.stem(y_int, label='$y_{int}[n]$', markerfmt='ro', linefmt='r-')
    plt.stem(s, label='$s[n]$', markerfmt='bo')
    plt.title('M = {}'.format(m))
    plt.xlabel('n')
    plt.ylabel('signal value')
    plt.legend()
    plt.show()
```