Task 1

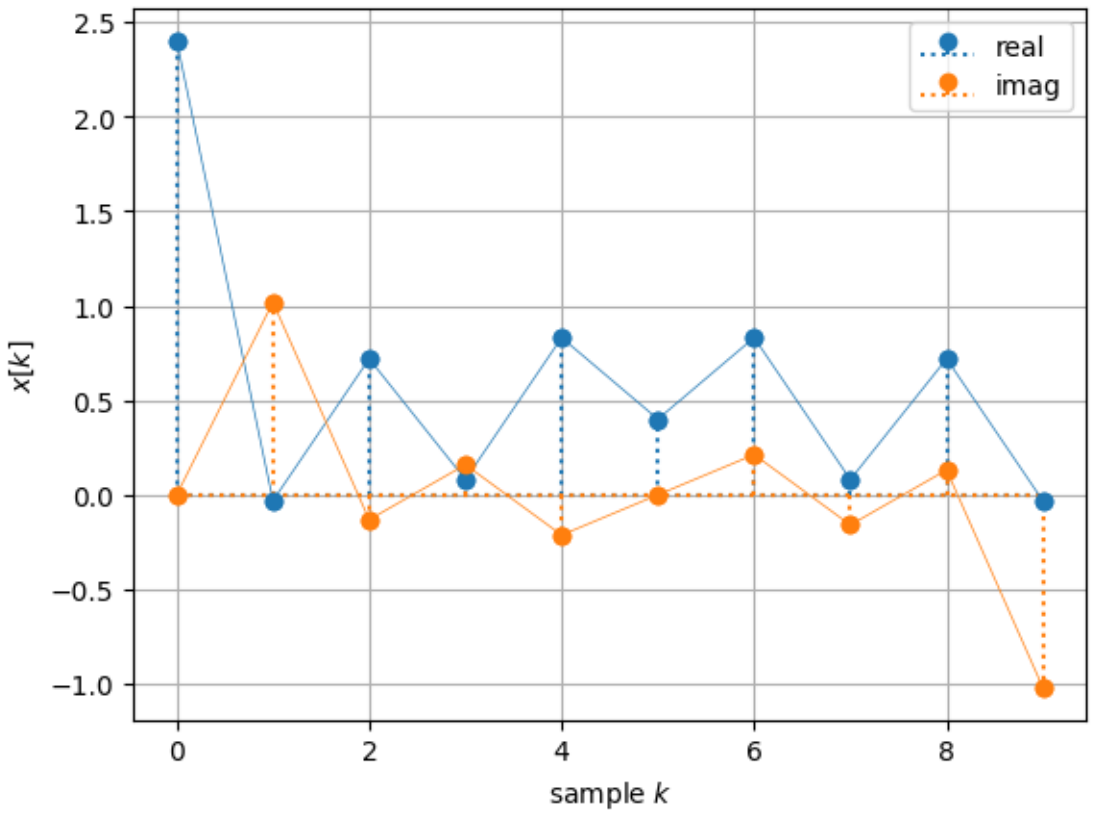
Mariusz Jagosz

October 3, 2024

[3]:
```
import numpy as np
import matplotlib.pyplot as plt
from numpy.linalg import inv
from numpy.fft import fft, ifft
```

[4]:
```
N = 10
k = np.arange(N)
mu = np.arange(N)


K = np.outer(k, mu)
W = np.exp(+1j * 2*np.pi/N * K)
X_test = np.array([6, 2, 4, 3, 4, 5, 0, 0, 0, 0])
x_test = 1/N * np.matmul(W, X_test)

plt.stem(k, np.real(x_test), label='real',
         markerfmt='C0o', basefmt='C0:', linefmt='C0:')
plt.stem(k, np.imag(x_test), label='imag',
         markerfmt='C1o', basefmt='C1:', linefmt='C1:')

plt.plot(k, np.real(x_test), 'C0o-', lw=0.5)
plt.plot(k, np.imag(x_test), 'C1o-', lw=0.5)
plt.xlabel(r'sample $k$')
plt.ylabel(r'$x[k]$')
plt.legend()
plt.grid(True)

print(np.allclose(ifft(X_test), x_test))
print('DC is 1 as expected: ', np.mean(x_test))
```

```
True
DC is 1 as expected:  (0.6+8.881784197001253e-17j)
```



[14]:
```
x_test2 = X_test[0] * W[:, 0] + X_test[1] * W[:, 1] + X_test[2] * W[:, 2]␣
↪+X_test[3] * W[:, 3]  + X_test[4] * W[:, 4] +  X_test[5] * W[:, 5]
```

[15]:
```
x_test2 *= 1/N
print(np.allclose(x_test, x_test2))
```

```
True
```

[12]:
```
K
```

[12]:
```
array([[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0],
       [ 0,  1,  2,  3,  4,  5,  6,  7,  8,  9],
       [ 0,  2,  4,  6,  8, 10, 12, 14, 16, 18],
       [ 0,  3,  6,  9, 12, 15, 18, 21, 24, 27],
       [ 0,  4,  8, 12, 16, 20, 24, 28, 32, 36],
       [ 0,  5, 10, 15, 20, 25, 30, 35, 40, 45],
       [ 0,  6, 12, 18, 24, 30, 36, 42, 48, 54],
       [ 0,  7, 14, 21, 28, 35, 42, 49, 56, 63],
       [ 0,  8, 16, 24, 32, 40, 48, 56, 64, 72],
       [ 0,  9, 18, 27, 36, 45, 54, 63, 72, 81]])
```

[29]:
```
W
```

[29]:
```
array([[ 1.        +0.00000000e+00j,  1.        +0.00000000e+00j,
         1.        +0.00000000e+00j,  1.        +0.00000000e+00j,
         1.        +0.00000000e+00j,  1.        +0.00000000e+00j,
         1.        +0.00000000e+00j,  1.        +0.00000000e+00j],
       [ 1.        +0.00000000e+00j,  0.80901699+5.87785252e-01j,
         0.30901699+9.51056516e-01j, -0.30901699+9.51056516e-01j,
        -0.80901699+5.87785252e-01j, -1.        +1.22464680e-16j,
        -0.80901699-5.87785252e-01j, -0.30901699-9.51056516e-01j,
         0.30901699-9.51056516e-01j,  0.80901699-5.87785252e-01j],
       [ 1.        +0.00000000e+00j,  0.30901699+9.51056516e-01j,
        -0.80901699+5.87785252e-01j, -0.80901699-5.87785252e-01j,
         0.30901699-9.51056516e-01j,  1.        -2.44929360e-16j,
         0.30901699+9.51056516e-01j, -0.80901699+5.87785252e-01j,
        -0.80901699-5.87785252e-01j,  0.30901699-9.51056516e-01j],
       [ 1.        +0.00000000e+00j, -0.30901699+9.51056516e-01j,
        -0.80901699-5.87785252e-01j,  0.80901699-5.87785252e-01j,
         0.30901699+9.51056516e-01j, -1.        +3.67394040e-16j,
         0.30901699-9.51056516e-01j,  0.80901699+5.87785252e-01j,
        -0.80901699+5.87785252e-01j, -0.30901699-9.51056516e-01j],
       [ 1.        +0.00000000e+00j, -0.80901699+5.87785252e-01j,
         0.30901699-9.51056516e-01j,  0.30901699+9.51056516e-01j,
        -0.80901699-5.87785252e-01j,  1.        -4.89858720e-16j,
        -0.80901699+5.87785252e-01j,  0.30901699-9.51056516e-01j,
         0.30901699+9.51056516e-01j, -0.80901699-5.87785252e-01j],
       [ 1.        +0.00000000e+00j, -1.        +1.22464680e-16j,
         1.        -2.44929360e-16j, -1.        +3.67394040e-16j,
         1.        -4.89858720e-16j, -1.        +6.12323400e-16j,
         1.        -7.34788079e-16j, -1.        +8.57252759e-16j,
         1.        -9.79717439e-16j, -1.        +1.10218212e-15j],
       [ 1.        +0.00000000e+00j, -0.80901699-5.87785252e-01j,
         0.30901699+9.51056516e-01j,  0.30901699-9.51056516e-01j,
        -0.80901699+5.87785252e-01j,  1.        -7.34788079e-16j,
        -0.80901699-5.87785252e-01j,  0.30901699+9.51056516e-01j,
         0.30901699-9.51056516e-01j, -0.80901699+5.87785252e-01j],
       [ 1.        +0.00000000e+00j, -0.30901699-9.51056516e-01j,
        -0.80901699+5.87785252e-01j,  0.80901699+5.87785252e-01j,
         0.30901699-9.51056516e-01j, -1.        +8.57252759e-16j,
         0.30901699+9.51056516e-01j,  0.80901699-5.87785252e-01j,
        -0.80901699-5.87785252e-01j, -0.30901699+9.51056516e-01j],
       [ 1.        +0.00000000e+00j,  0.30901699-9.51056516e-01j,
        -0.80901699-5.87785252e-01j, -0.80901699+5.87785252e-01j,
         0.30901699+9.51056516e-01j,  1.        -9.79717439e-16j,
         0.30901699-9.51056516e-01j, -0.80901699-5.87785252e-01j,
        -0.80901699+5.87785252e-01j,  0.30901699+9.51056516e-01j],
```

```
[ 1.        +0.00000000e+00j,  0.80901699-5.87785252e-01j,
  0.30901699-9.51056516e-01j, -0.30901699-9.51056516e-01j,
 -0.80901699-5.87785252e-01j, -1.        +1.10218212e-15j,
 -0.80901699+5.87785252e-01j, -0.30901699+9.51056516e-01j,
  0.30901699+9.51056516e-01j,  0.80901699+5.87785252e-01j]])
```

[44]:
```
import pandas as pd
df = pd.DataFrame(K)
```

[42]:
```
latex_code = df.to_latex(header = False, index=False)
print(latex_code)
```

```
\begin{tabular}{rrrrrrrrrr}
\toprule
\midrule
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 \\
0 & 2 & 4 & 6 & 8 & 10 & 12 & 14 & 16 & 18 \\
0 & 3 & 6 & 9 & 12 & 15 & 18 & 21 & 24 & 27 \\
0 & 4 & 8 & 12 & 16 & 20 & 24 & 28 & 32 & 36 \\
0 & 5 & 10 & 15 & 20 & 25 & 30 & 35 & 40 & 45 \\
0 & 6 & 12 & 18 & 24 & 30 & 36 & 42 & 48 & 54 \\
0 & 7 & 14 & 21 & 28 & 35 & 42 & 49 & 56 & 63 \\
0 & 8 & 16 & 24 & 32 & 40 & 48 & 56 & 64 & 72 \\
0 & 9 & 18 & 27 & 36 & 45 & 54 & 63 & 72 & 81 \\
\bottomrule
\end{tabular}
```

[33]:
```
df_w = pd.DataFrame(W)
```

[43]:
```
latex_code = df_w.to_latex(header = False, index=False)
print(latex_code)
```

```
\begin{tabular}{rrrrrrrrrr}
\toprule
\midrule
1.000000+0.000000j & 1.000000+0.000000j & 1.000000+0.000000j &
1.000000+0.000000j & 1.000000+0.000000j & 1.000000+0.000000j &
1.000000+0.000000j & 1.000000+0.000000j & 1.000000+0.000000j &
1.000000+0.000000j \\
1.000000+0.000000j & 0.809017+0.587785j & 0.309017+0.951057j &
-0.309017+0.951057j & -0.809017+0.587785j & -1.000000+0.000000j &
-0.809017-0.587785j & -0.309017-0.951057j & 0.309017-0.951057j &
0.809017-0.587785j \\
1.000000+0.000000j & 0.309017+0.951057j & -0.809017+0.587785j &
-0.809017-0.587785j & 0.309017-0.951057j & 1.000000-0.000000j &
0.309017+0.951057j & -0.809017+0.587785j & -0.809017-0.587785j &
```

```
0.309017-0.951057j \\
1.000000+0.000000j & -0.309017+0.951057j & -0.809017-0.587785j &
0.809017-0.587785j & 0.309017+0.951057j & -1.000000+0.000000j &
0.309017-0.951057j & 0.809017+0.587785j & -0.809017+0.587785j &
-0.309017-0.951057j \\
1.000000+0.000000j & -0.809017+0.587785j & 0.309017-0.951057j &
0.309017+0.951057j & -0.809017-0.587785j & 1.000000-0.000000j &
-0.809017+0.587785j & 0.309017-0.951057j & 0.309017+0.951057j &
-0.809017-0.587785j \\
1.000000+0.000000j & -1.000000+0.000000j & 1.000000-0.000000j &
-1.000000+0.000000j & 1.000000-0.000000j & -1.000000+0.000000j &
1.000000-0.000000j & -1.000000+0.000000j & 1.000000-0.000000j &
-1.000000+0.000000j \\
1.000000+0.000000j & -0.809017-0.587785j & 0.309017+0.951057j &
0.309017-0.951057j & -0.809017+0.587785j & 1.000000-0.000000j &
-0.809017-0.587785j & 0.309017+0.951057j & 0.309017-0.951057j &
-0.809017+0.587785j \\
1.000000+0.000000j & -0.309017-0.951057j & -0.809017+0.587785j &
0.809017+0.587785j & 0.309017-0.951057j & -1.000000+0.000000j &
0.309017+0.951057j & 0.809017-0.587785j & -0.809017-0.587785j &
-0.309017+0.951057j \\
1.000000+0.000000j & 0.309017-0.951057j & -0.809017-0.587785j &
-0.809017+0.587785j & 0.309017+0.951057j & 1.000000-0.000000j &
0.309017-0.951057j & -0.809017-0.587785j & -0.809017+0.587785j &
0.309017+0.951057j \\
1.000000+0.000000j & 0.809017-0.587785j & 0.309017-0.951057j &
-0.309017-0.951057j & -0.809017-0.587785j & -1.000000+0.000000j &
-0.809017+0.587785j & -0.309017+0.951057j & 0.309017+0.951057j &
0.809017+0.587785j \\
\bottomrule
\end{tabular}
```


[ ]: