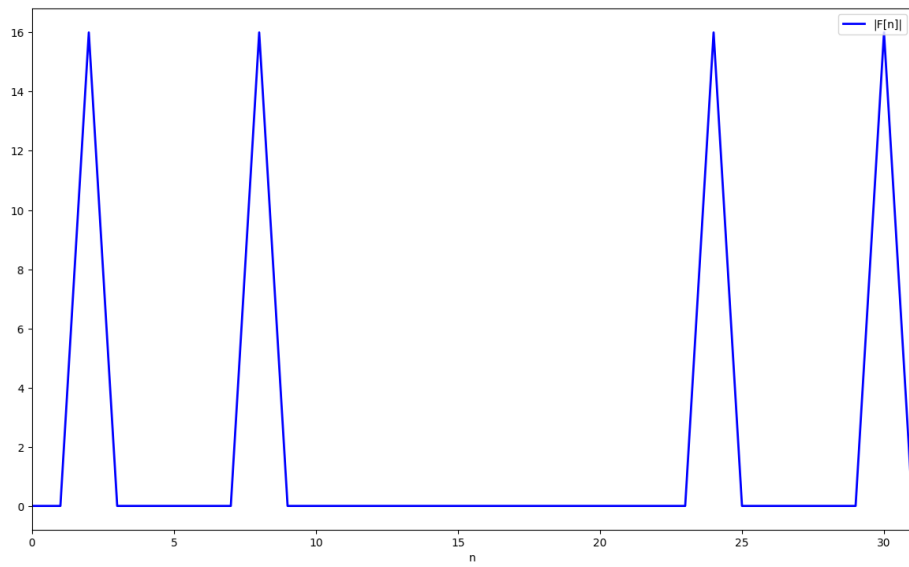
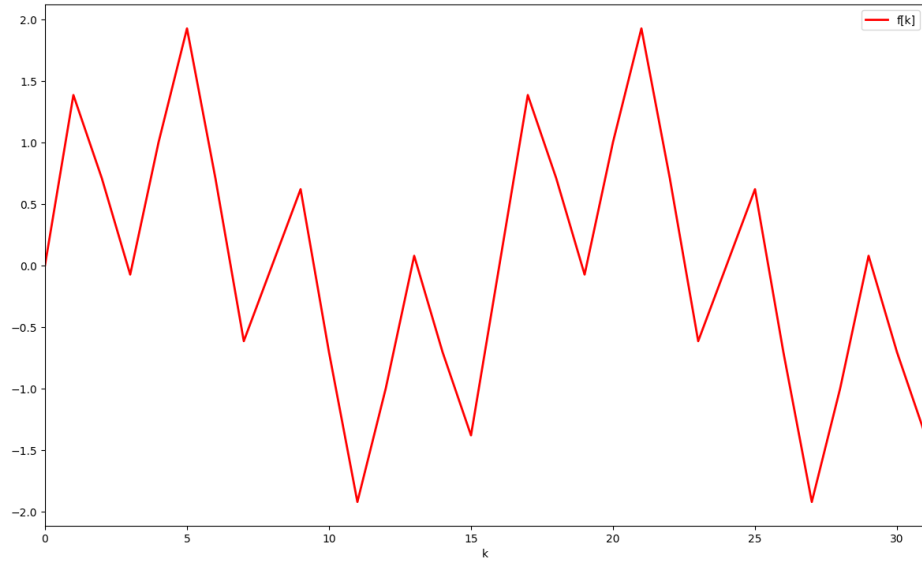


デジタル信号処理 第9回宿題

氏名：園山佳典

学籍番号：26002201991



## ソースコード

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

# パラメータ
T = 1
w1 = 0.125 * np.pi / T
w2 = 0.5 * np.pi / T
L = 32
N = 32

# k=0,1,2,...,L-1 のインデックスを作る
k = np.arange(start=0, stop=L, step=1)

# f[k] = sin(w1kT)の信号を作成
f = np.sin(w1 * k * T) + np.sin(w2 * k * T)

# DFT 行列を計算
n = np.arange(start=0, stop=N, step=1)
w0 = np.exp(-1j * 2 * np.pi / N)
mtx_dft = np.zeros((N, N)) + 1j * np.zeros((N, N))

for i in range(N):
    for j in range(N):
        mtx_dft[i, j] = w0 ** (i * j)

# f[k]の長さがNより短い場合、ゼロを詰める
f_fft = f
f_ex = np.zeros((N - f.size, 1))
if not len(f_ex) == 0:
    f_fft = np.concatenate(f, f_ex)

# 行列演算でDFTを行う(9回目の講義資料に参考)
F = np.dot(mtx_dft, f_fft)
```

```
# スペクトル F[n]は複素数
# 絶対値で振幅スペクトル|F[n]|を計算
F_amp = abs(F)

# f[k]の波形を描画
plt.figure()
# 横軸 k、縦軸 f[k]で描画
# 線の色を赤、太さを2に設定
plt.plot(k, f, color="r", linewidth=2, label="f[k]")
# 横軸の範囲を 0~L-1 に設定
plt.xlim((0, L - 1))
# 横軸のラベルを k に設定
plt.xlabel("k")
# 凡例
plt.legend(loc="upper right")
plt.show()

# F[n]の波形を描画
plt.figure()
# 横軸 n、縦軸 F_amp[n]で描画
# 線の色を青、太さを2に設定
plt.plot(n, F_amp, color="b", linewidth=2, label="|F[n]|")
# 横軸の範囲を 0~N-1 に設定
plt.xlim((0, N - 1))
# 横軸のラベルを k に設定
plt.xlabel("n")
# 凡例
plt.legend(loc="upper right")
plt.show()
```