

# Laboratorium 1

Biblioteki: numpy, matplotlib

Celem laboratorium 1. jest poznanie podstaw cyfrowego przetwarzania sygnałów — przeprowadzenie syntetycznej akwizycji sygnału, jego normalizacji i kwantyzacji oraz przeprowadzenie prostej procedury odszumienia sygnału.

## Zadanie 1:

- Zainicjalizować wykres (3 kolumny x 3 wiersze) o wymiarach 10 x 10 cali.
- Spróbować funkcję sinus w zakresie  $0-4\pi$  w 40 kwantach.

W tym celu należy zadeklarować wartości z przedziału X (zmienną nazwać `x`) poleceniem `np.linspace` w podanym zakresie, o podanej liczbie kwantów.

Następnie dla każdej z wartości wywołać funkcję sinus: `y = np.sin(x)`.

- W pierwszej komórce wykresu narysować spróbkowaną funkcję przy użyciu polecenia `plot`.
- Utworzyć dwuwymiarowe złożenie spróbkowanej funkcji. Ponieważ y jest jednowymiarowym wektorem należy użyć `np.newaxis` do adresacji w nowym wymiarze:

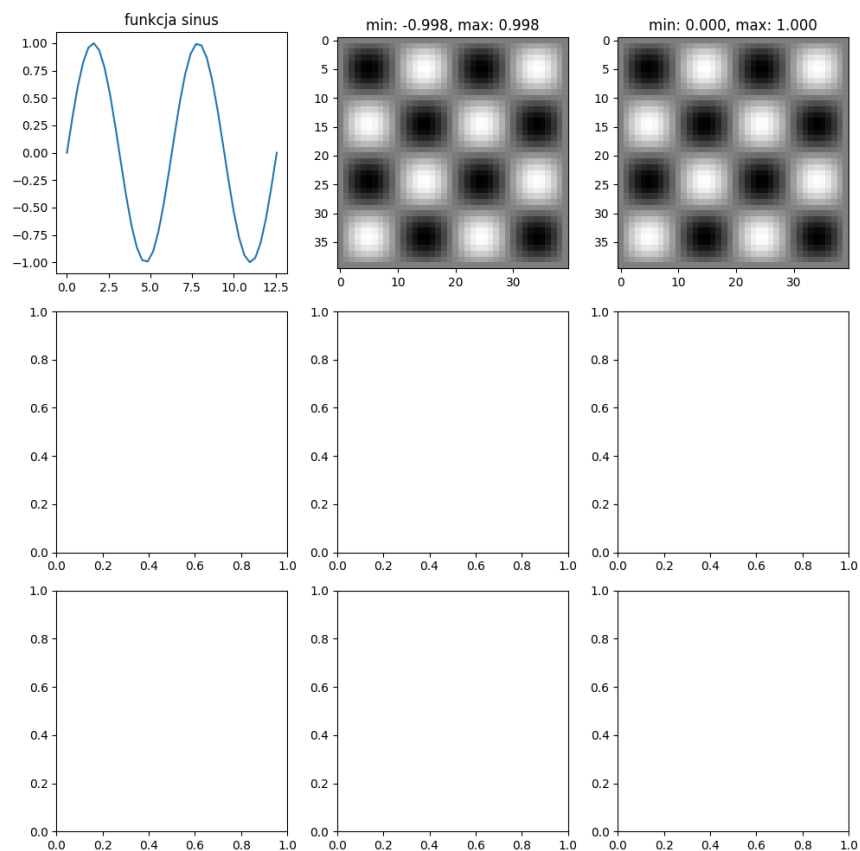
```
y[:, np.newaxis]*y[np.newaxis, :]
```

### ▼ Ciekawostka!

W miejsce `newaxis` można użyć `None`.

- Otrzymany obraz narysować w drugiej komórce wykresu. Użyć cmapy *binary*. W podpisie umieścić maksymalną i minimalną wartość intensywności obrazu, wyznaczone przy użyciu funkcji `np.min` oraz `np.max`, z dokładnością do 3 miejsc po przecinku.
- Wykonać normalizację przedziałową obrazu do zakresu 0-1. W tym celu od każdego piksela obrazu należy odjąć wartość minimalną, następnie podzielić przez wartość maksymalną w całej macierzy.
- Znormalizowany obraz, wraz z podpisem zawierającym wartość minimalną i maksymalną, umieścić w kolejnej komórce pierwszego wiersza wykresu.

## Efekt zadania 1:

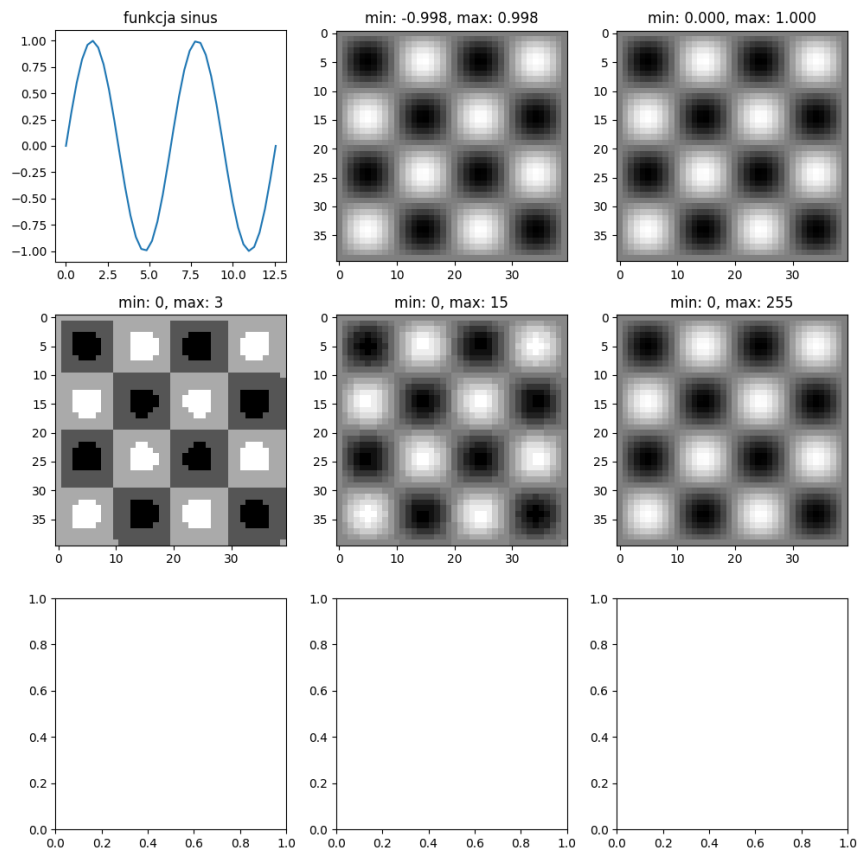


## Zadanie 2:

Kolejnym etapem jest dyskretyzacja otrzymanego po normalizacji obrazu.

- W drugim wierszu narysować kolejno obrazy o głębi 2, 4 i 8-bitowej. Do wygenerowania takiego obrazu należy w pierwszej kolejności obliczyć maksymalną wartość piksela w danej reprezentacji oraz – do zaokrąglenia do wartości całkowitoliczbowej – użyć funkcji `np rint`.
- W podpisach wyświetlić wartość minimalną i maksymalną.

## Efekt zadania 2:



### Zadanie 3:

- Do znormalizowanego obrazu dodać szum z rozkładu normalnego, wygenerowanego przy użyciu funkcji `np.random.normal`.
- Efekt narysować w pierwszej kolumnie i trzecim wierszu wykresu.
- Następnie należy wygenerować `n=[50,1000]` zaszumionych rozkładem normalnym obrazów. Odszumienie obrazu uzyskamy dzięki zsumowaniu  $n$  zaszumionych obrazów, i następnie ich normalizacji.
- Wynik odszumienia przy użyciu 50 i 1000 obrazów narysować w ostatnich dwóch komórkach wykresu.

Efekt zadania 3:

