

TECHNIKI PROGRAMOWANIA - projekt 3

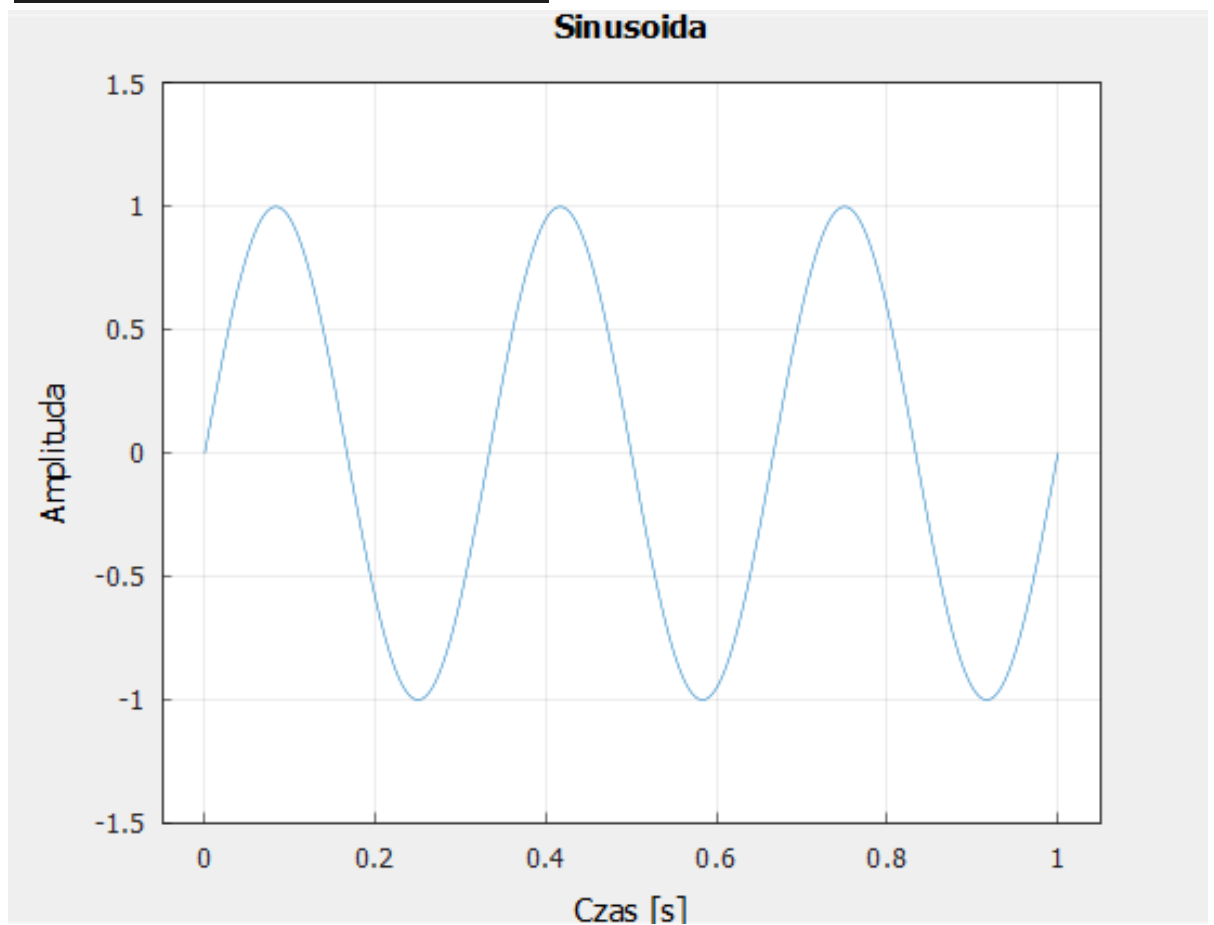
Tymoteusz Żochowski 203840

Mikołaj Cyrzan 203193

1. Generowanie i wyświetlanie sygnału

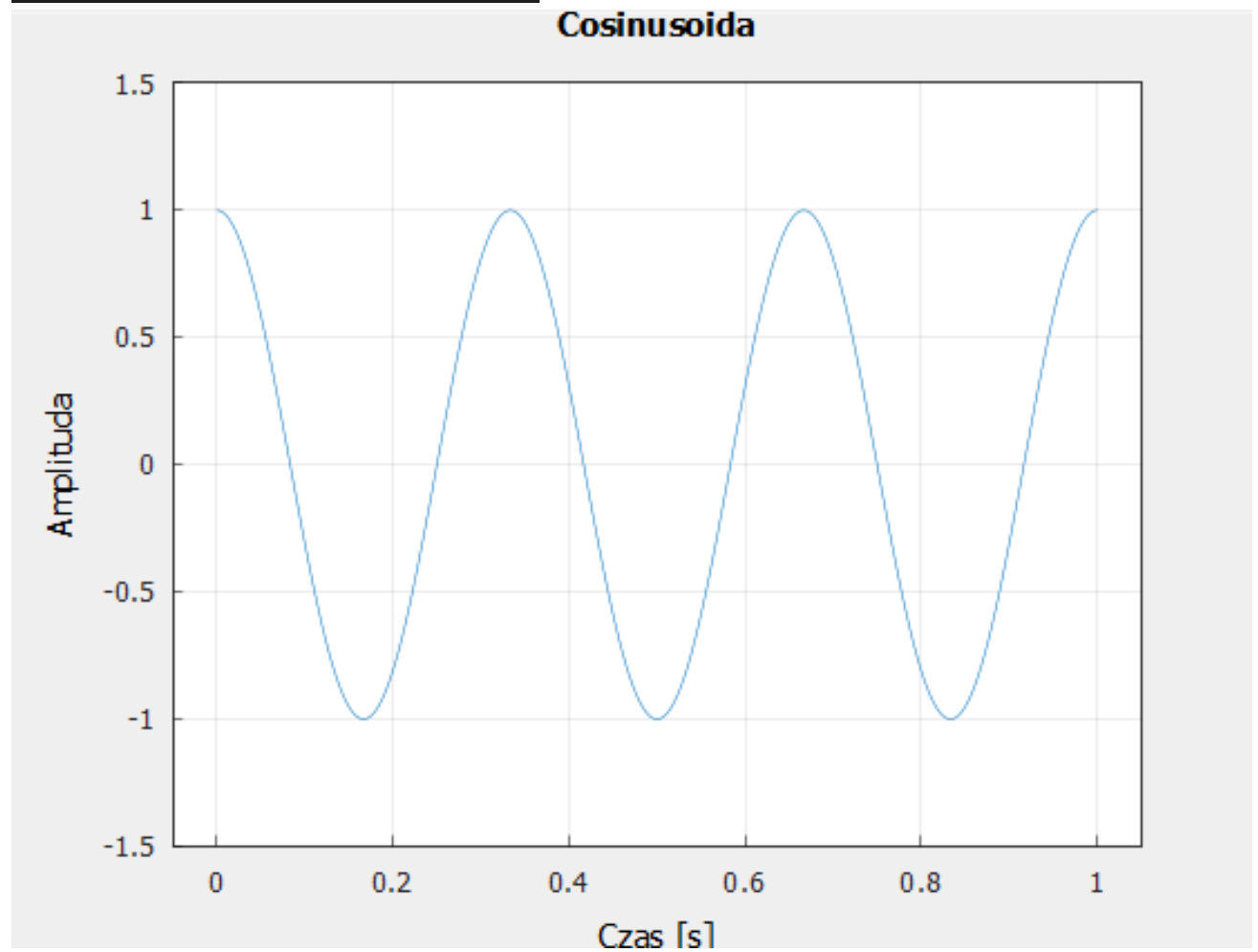
a)

```
#Sygnał Sinus  
data=biblioteka.sinus(1000,0,1,3)  
biblioteka.draw(data,"Sinusoida")
```



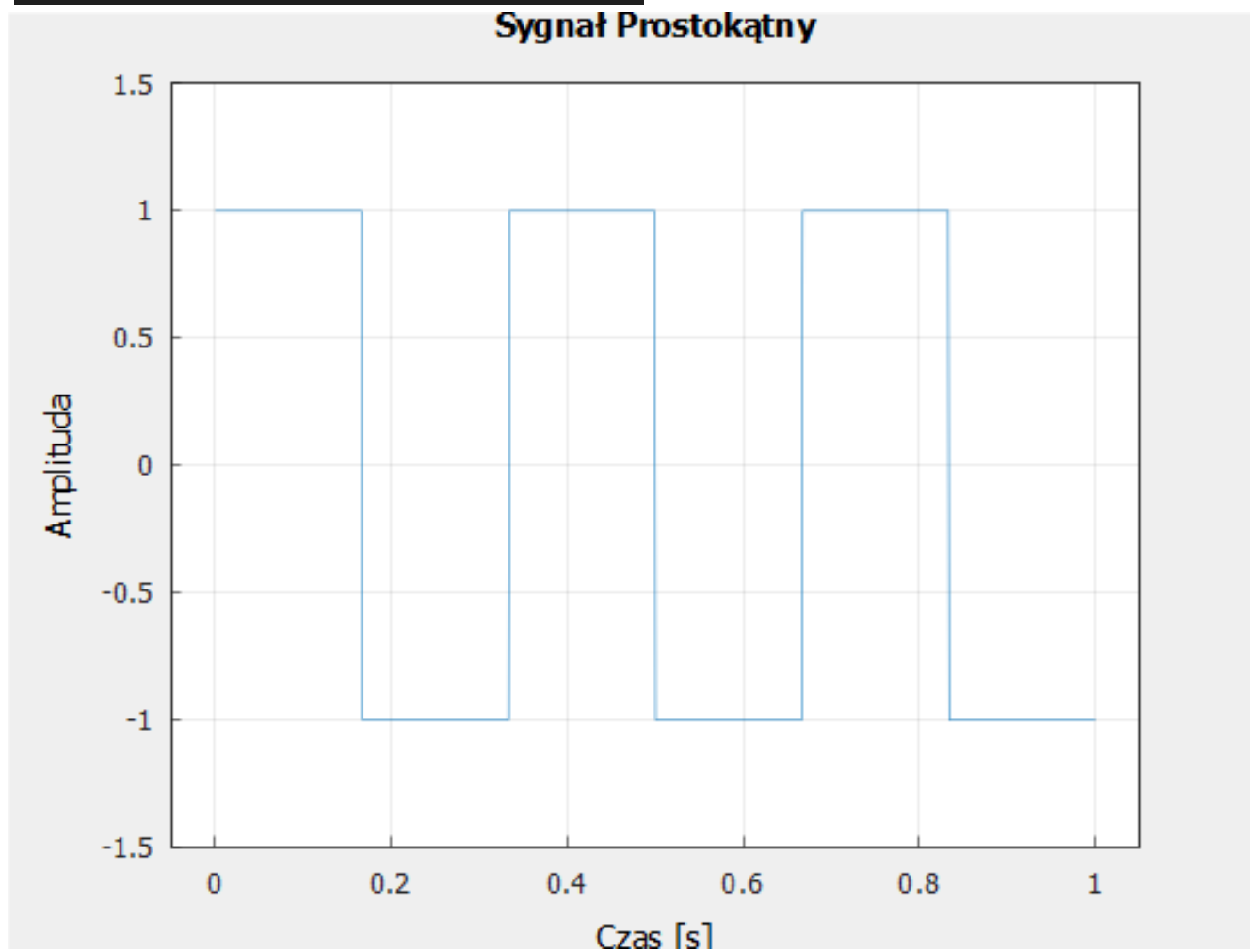
b)

```
#Sygnał Cosinus  
data=biblioteka.cosinus(1000,0,1,3)  
biblioteka.draw(data,"Cosinusoida")
```



c)

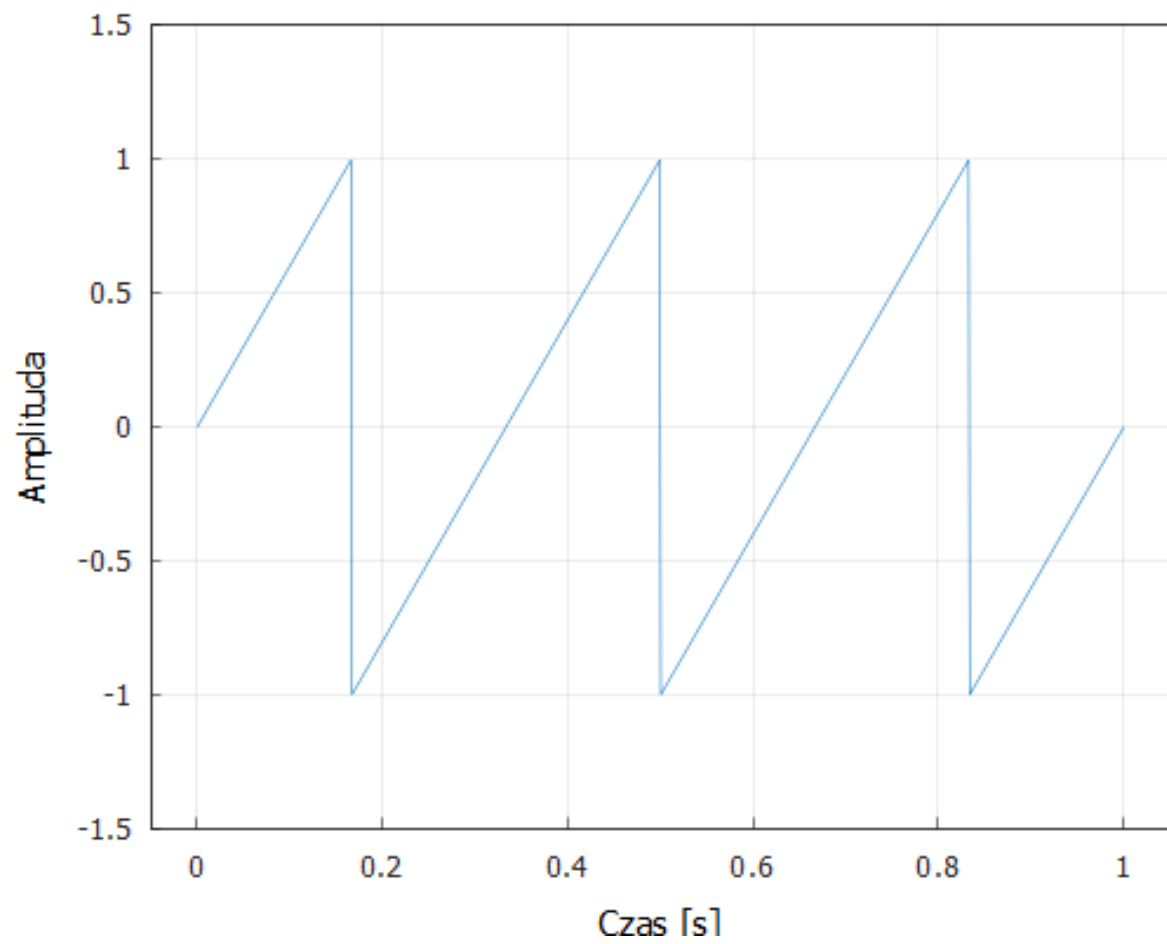
```
#Sygnał Prostokątny  
data=biblioteka.rectangle(1000,0,1,3)  
biblioteka.draw(data,"Sygnał Prostokątny")
```



d)

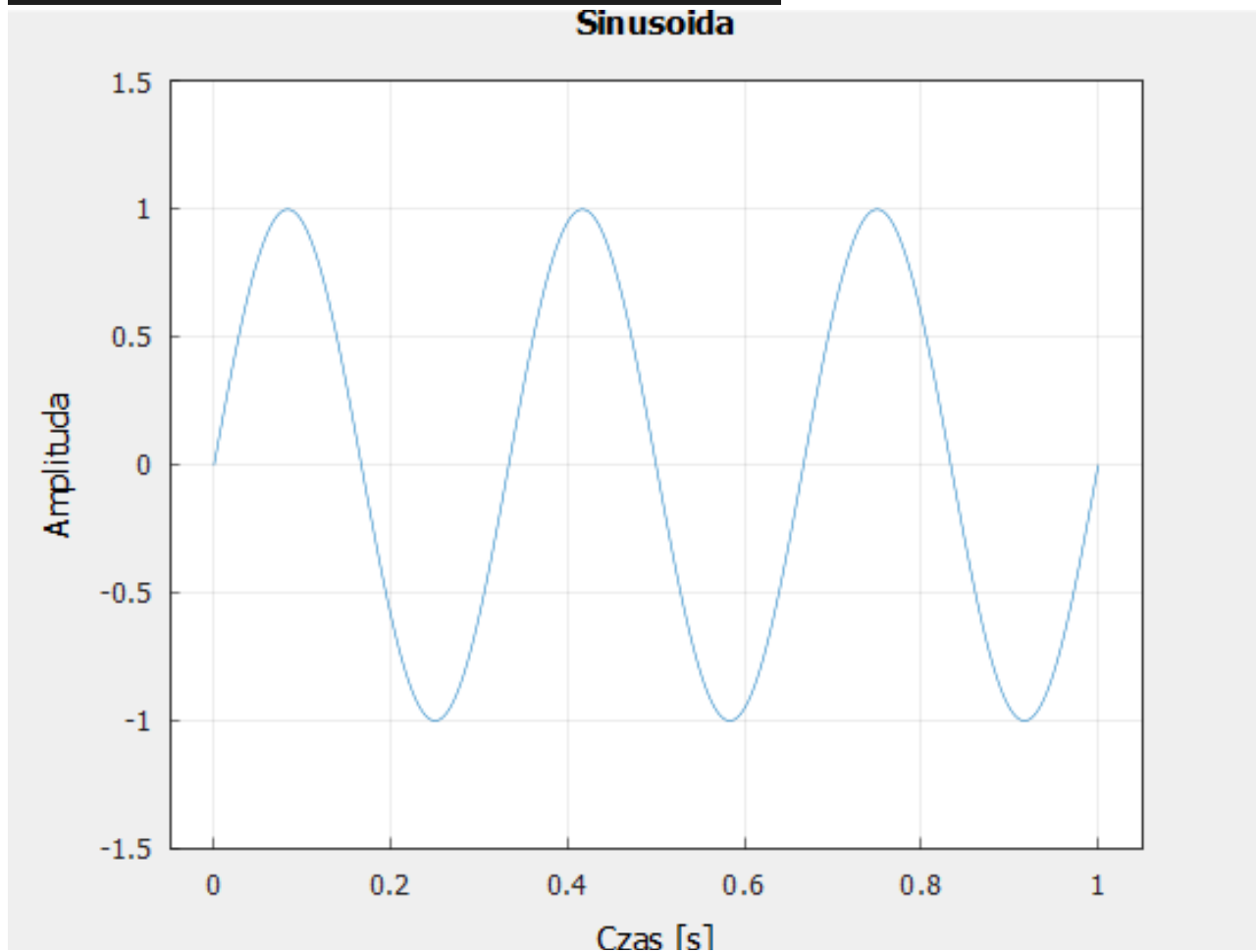
```
#Sygnał Piłkoształtny  
data=biblioteka.saw(1000,0,1,3)  
biblioteka.draw(data,"Sygnał Piłkoształtny")
```

Sygnał Piłkoształtny

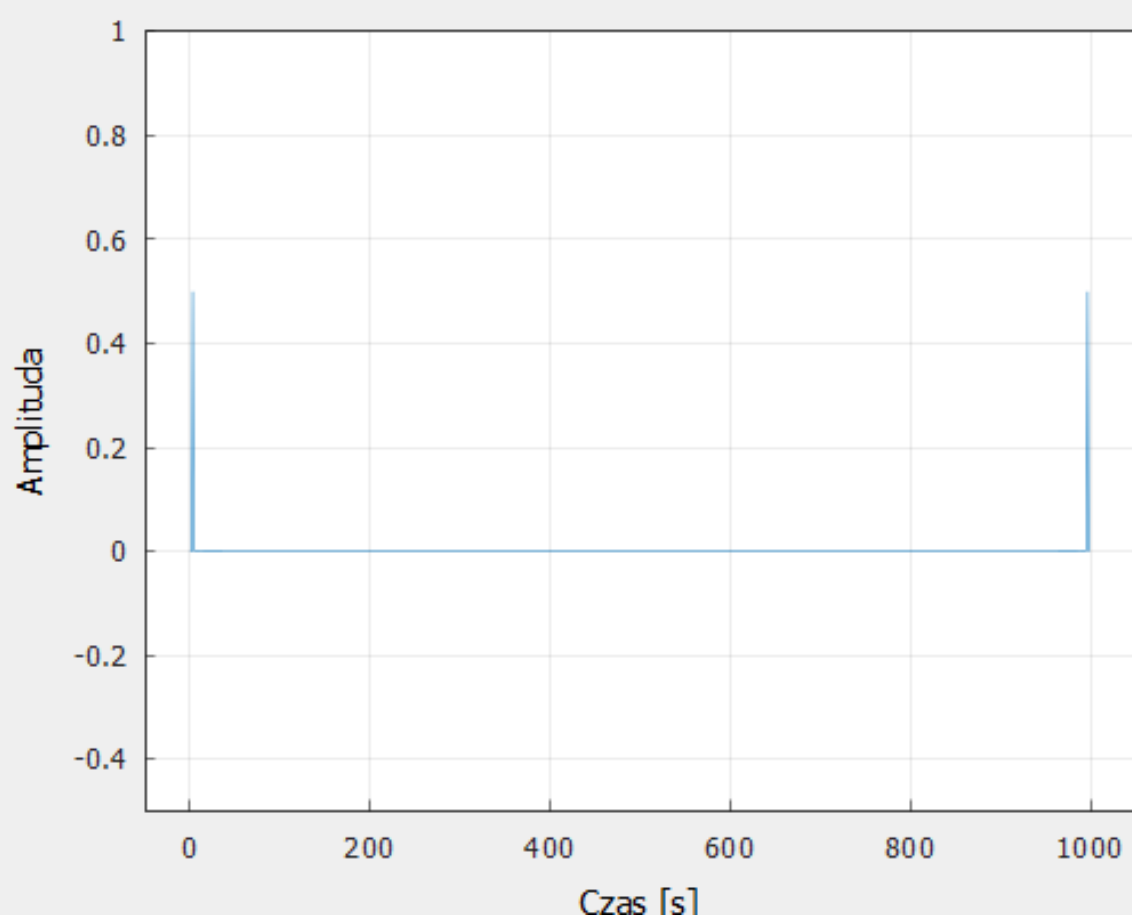


2. DFT oraz IDFT sygnału

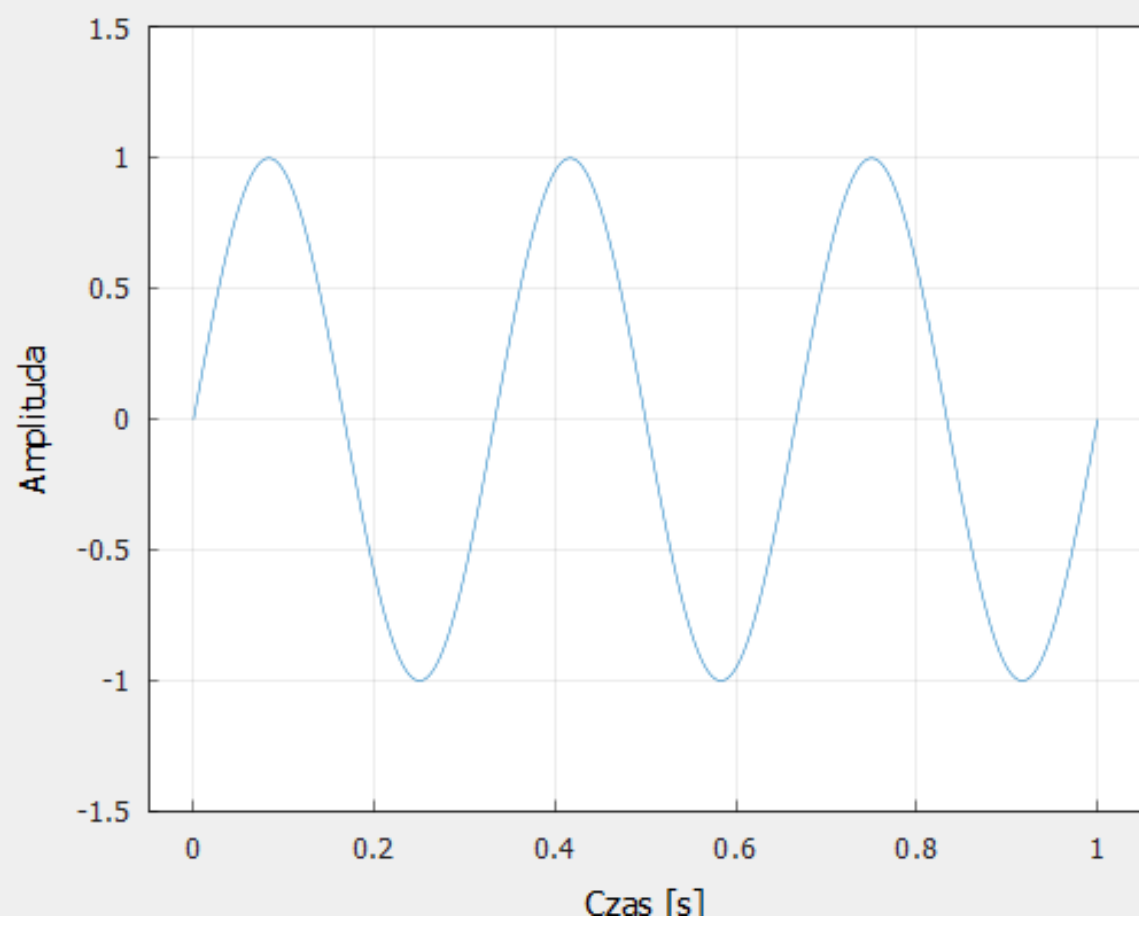
```
#DFT oraz IDFT sygnału (np.Sinus)
data=biblioteka.sinus(1000,0,1,3)
biblioteka.draw(data,"Sinusoida")
dft=biblioteka.DFT(data)
biblioteka.draw(biblioteka.spectrumDFT(dft),"DFT")
biblioteka.draw(biblioteka.IDFT(dft,0,1),"IDFT")
```



DFT



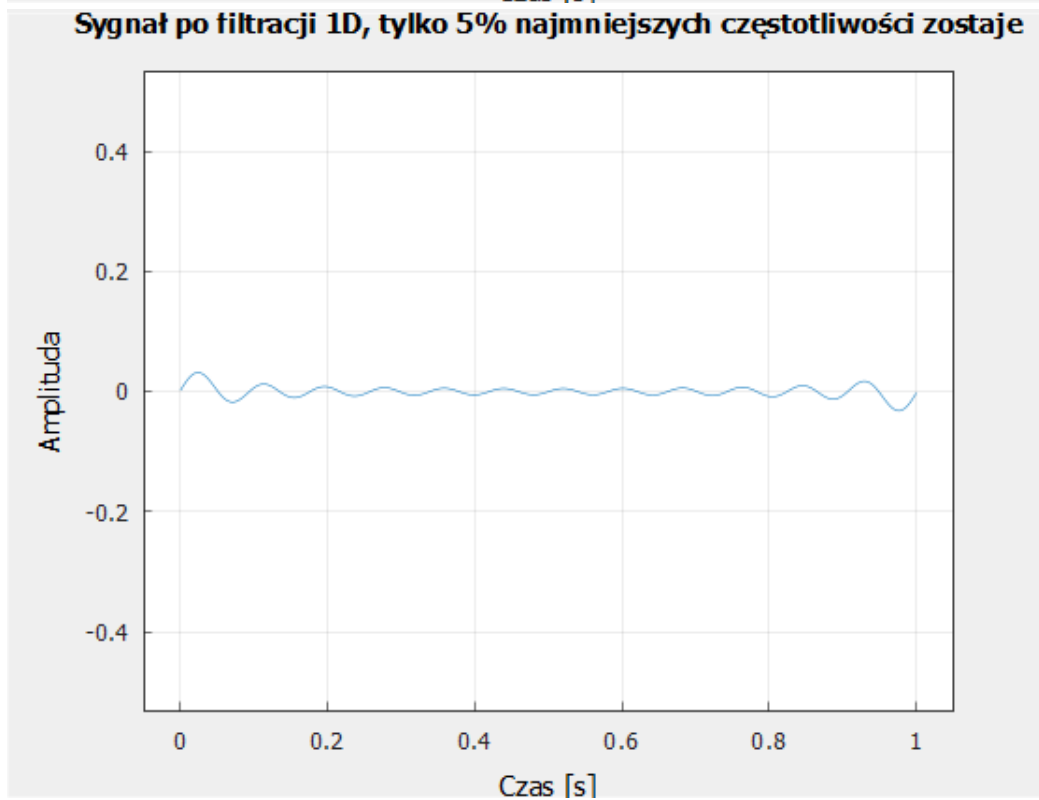
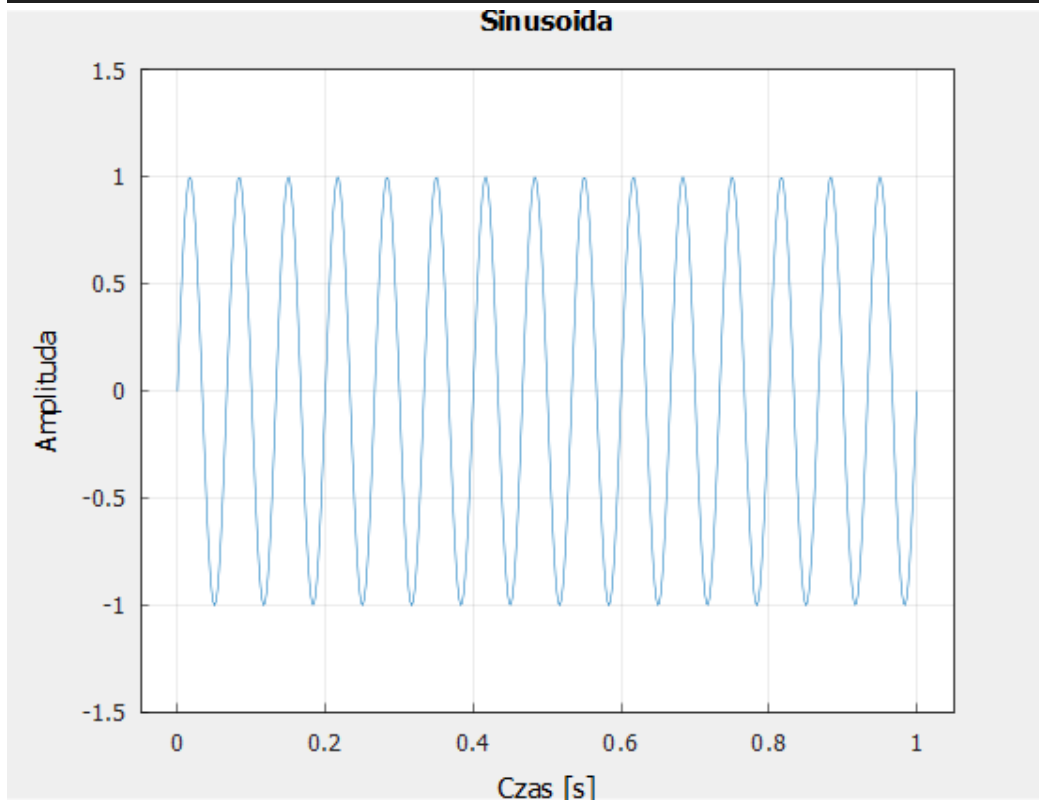
IDFT



3. Filtracja 1D oraz 2D

a) Filtracja 1D

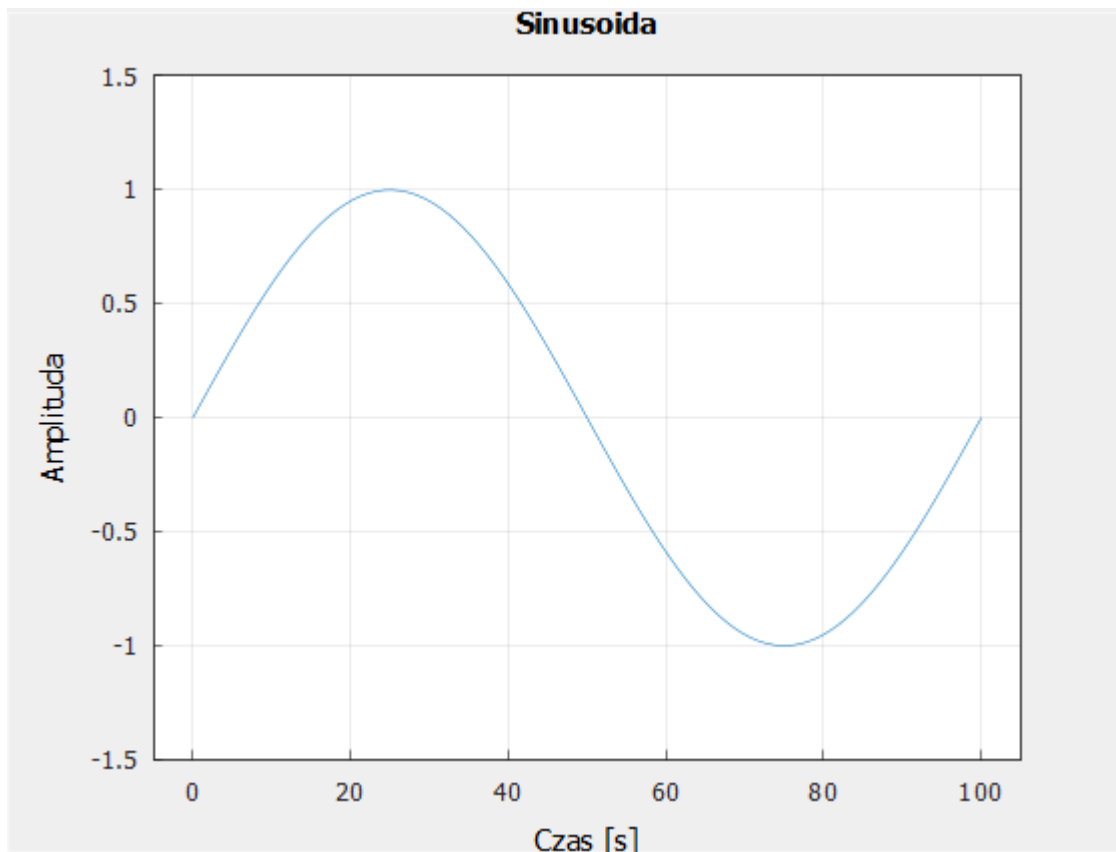
```
#Filtracja 1D sygnału (np.Sinus)
data=biblioteka.sinus(500,0,1,15)
biblioteka.draw(data,"Sinusoida")
dft=biblioteka.DFT(data)
filteredSignal1D=biblioteka.filter1D(dft,0.05)
biblioteka.draw(biblioteka.IDFT(filteredSignal1D,0,1),"Sygnał po filtracji 1D, tylko 5% najmniejszych częstotliwości zostaje")
```



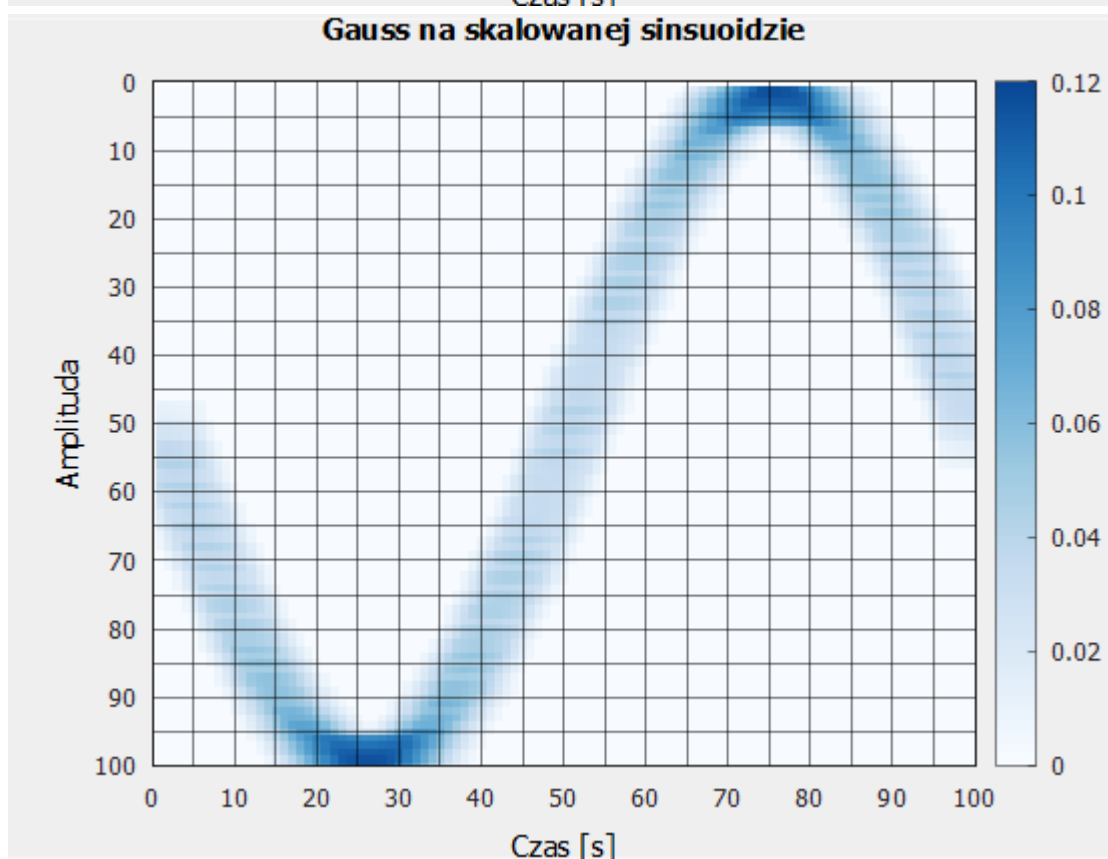
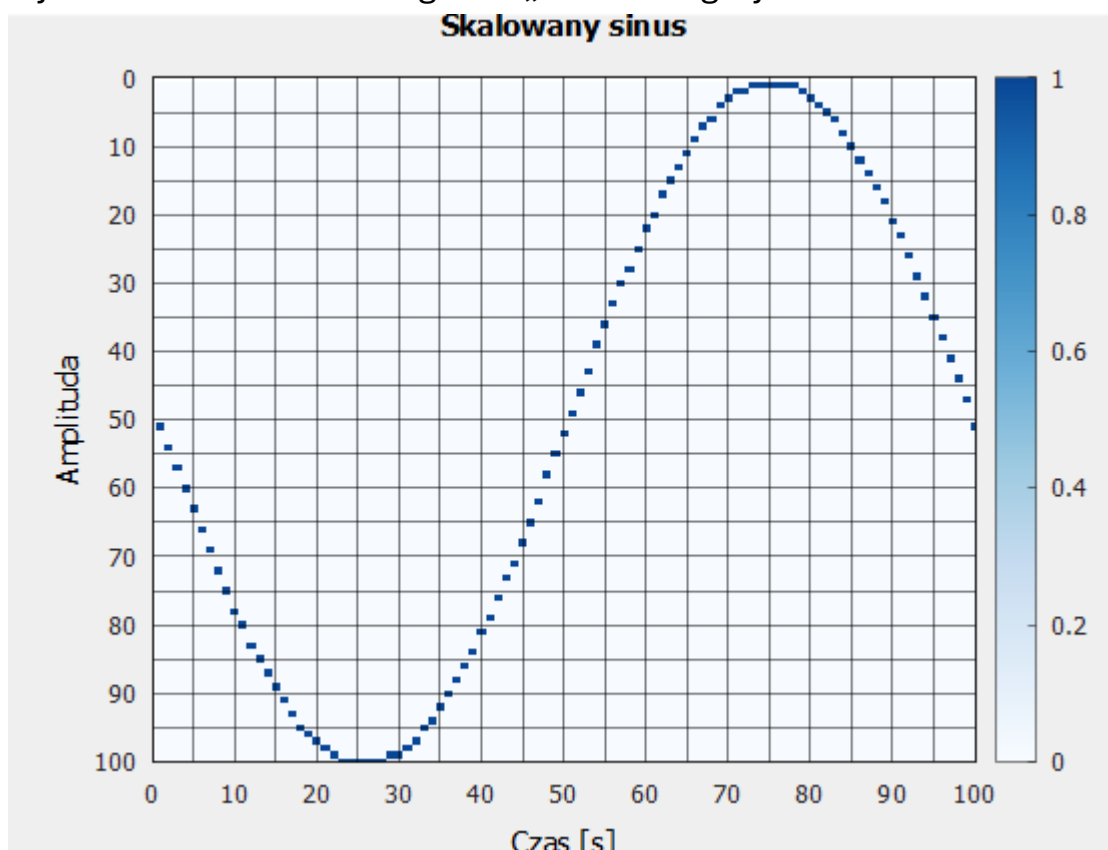
b) Filtracja 2D jako Rozmycie Gaussa (dla sinusa)

1. Wykres zwykłej sinusoidy
2. Przeskalowana sinusoida przedstawiona jako punkty
3. Rozmycie Gaussa (filtracja 2D)

```
#Filtracja 2D sygnału jako rozmycie Gaussa (np.Sinus)
data=biblioteka.sinus(100,0,100,1)
biblioteka.draw(data,"Sinusoida")
biblioteka.plotMatrix(biblioteka.convert1Dto2D(data,100),"Skalowany sinus")
biblioteka.plotMatrix(biblioteka.applyGaussianBlur2D(data, 10, 5, 100),"Gauss na skalowanej sinusoidzie")
```



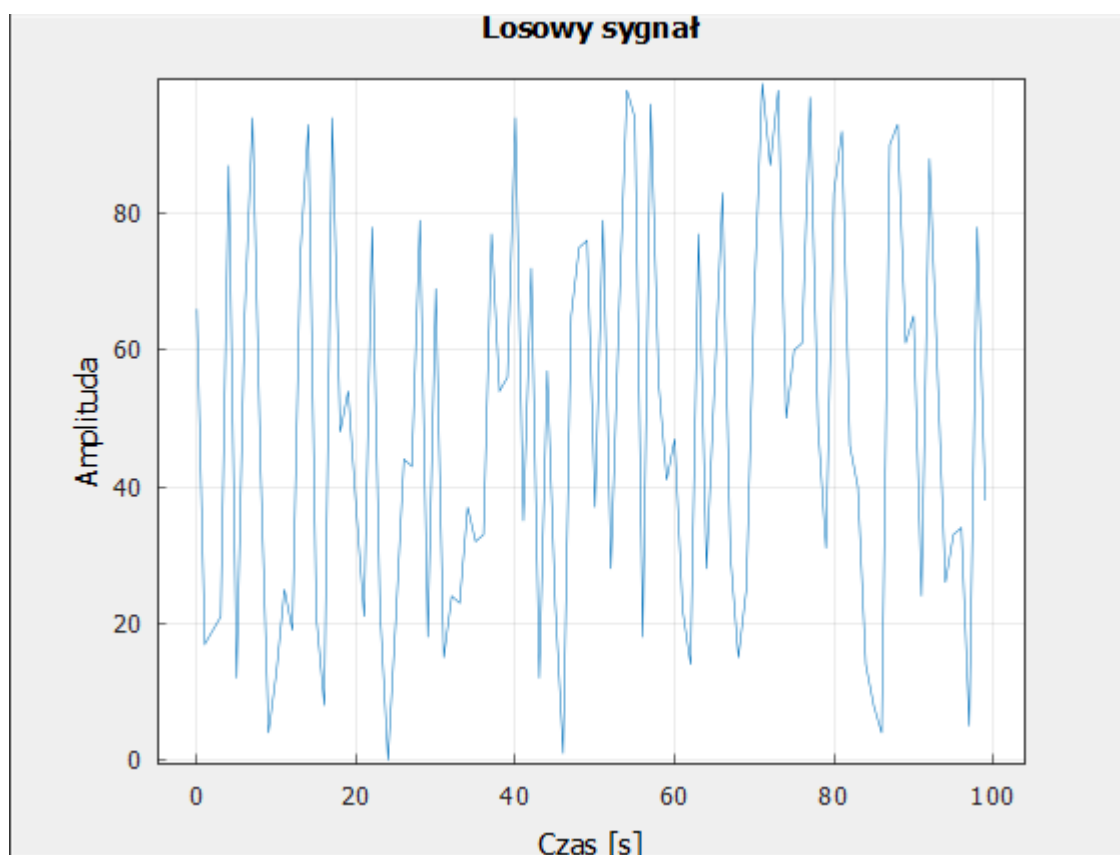
Uwaga: Sygnał wygląda na „odwrócony” gdyż podziałka podczas wyświetlania macierzowego ma „0” w OY u góry.



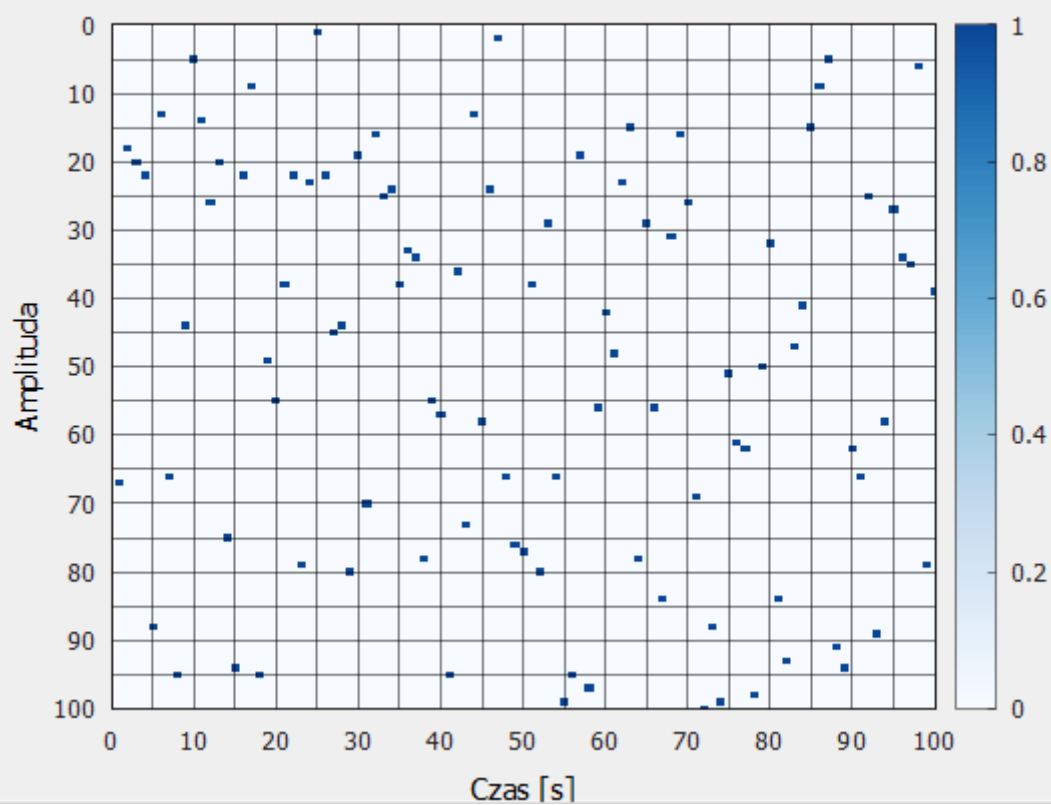
c) Filtracja 2D jako Rozmycie Gaussa (dla losowego sygnału)

1. Wykres losowo wygenerowanego sygnału
2. Losowy sygnał przedstawiony jako punkty
3. Rozmycie Gaussa (filtracja 2D)

```
#Filtracja 2D sygnału jako rozmycie Gaussa (Sygnał generowany losowo)
data=biblioteka.generateMatrix(100,100)
biblioteka.draw(data,"Losowy sygnał")
biblioteka.plotMatrix(biblioteka.convert1Dto2D(data,0),"Losowy sygnał jako punkty")
biblioteka.plotMatrix(biblioteka.applyGaussianBlur2D(data,100,1.5,0),"Gauss na losowym sygnale")
```



Losowy sygnał jako punkty



Gauss na losowym sygnale

