

# Techniki Programowania – projekt 3

Błażej Kiwacz 203407

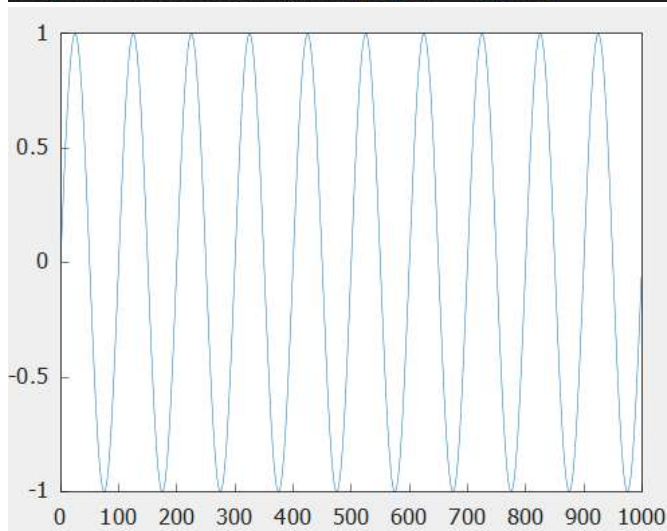
Filip Treder 203524

## 1) Generacja i wizualizacja zadanych sygnałów.

Jako oś x przyjęta jest dana próbka a jako oś y przyjęta jest amplituda sygnału.

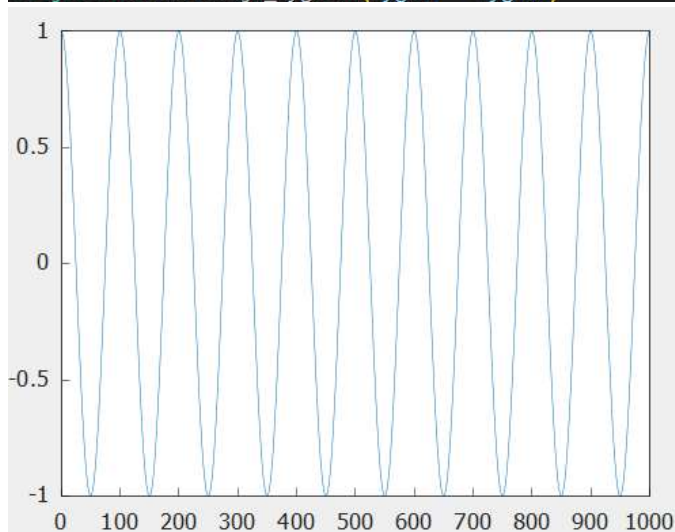
- Sygnał sinusoidalny.

```
sygnał = Projekt3.generuj_sin(czestotliwosc=10, czestotliwosc_probek=1000, czas_trwania=1.0)  
Projekt3.wizualizacja_sygnału(sygnał = sygnał)
```



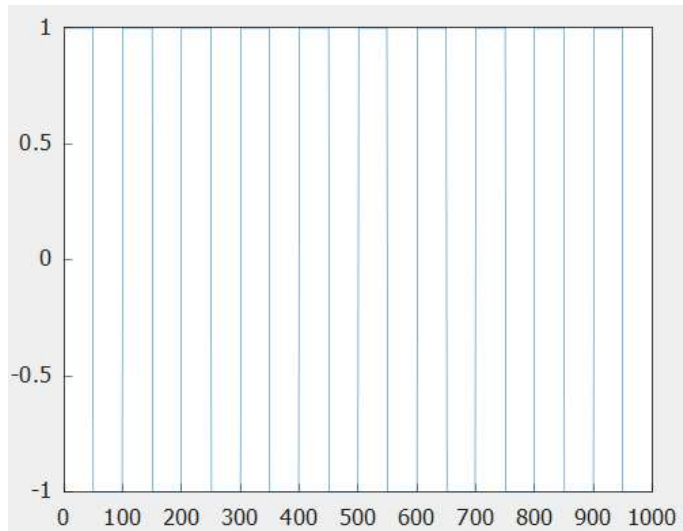
- Sygnał cosinusoidalny.

```
sygnał = Projekt3.generuj_cos(czestotliwosc=10, czestotliwosc_probek=1000, czas_trwania=1.0)  
Projekt3.wizualizacja_sygnału(sygnał = sygnał)
```



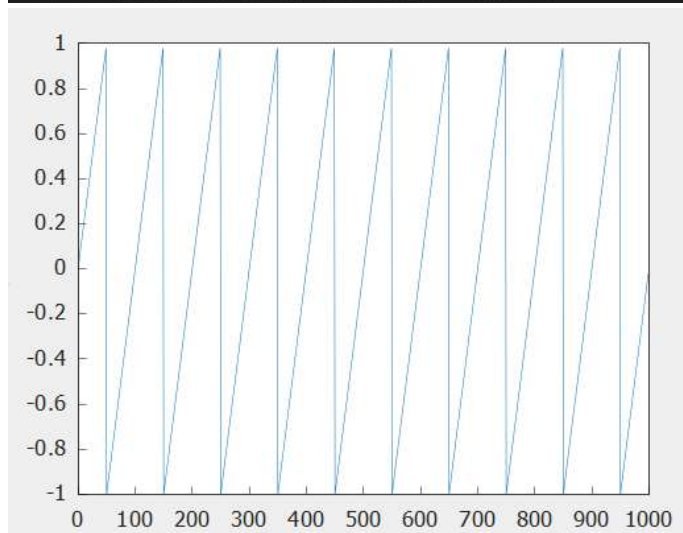
- Sygnał prostokątny.

```
sygnal = Projekt3.generuj_prostokat(czestotliwosc=10, czestotliwosc_probek=1000, czas_trwania=1.0)  
Projekt3.wizualizacja_sygnalu(sygnal = sygnal)
```



- Sygnał piłokształtny.

```
sygnal = Projekt3.generuj_pila(czestotliwosc=10, czestotliwosc_probek=1000, czas_trwania=1.0)  
Projekt3.wizualizacja_sygnalu(sygnal = sygnal)
```

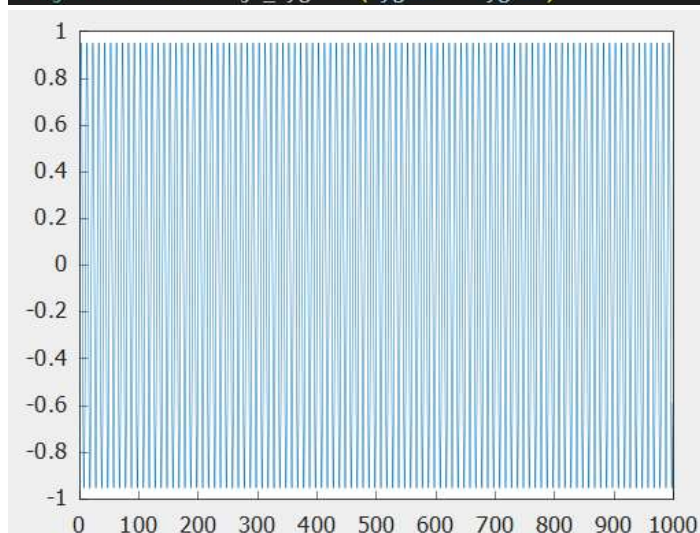


## 2) DFT i transformata odwrotna.

Najpierw generujemy sygnał który będziemy używać do DFT.

- Generacja sygnału.

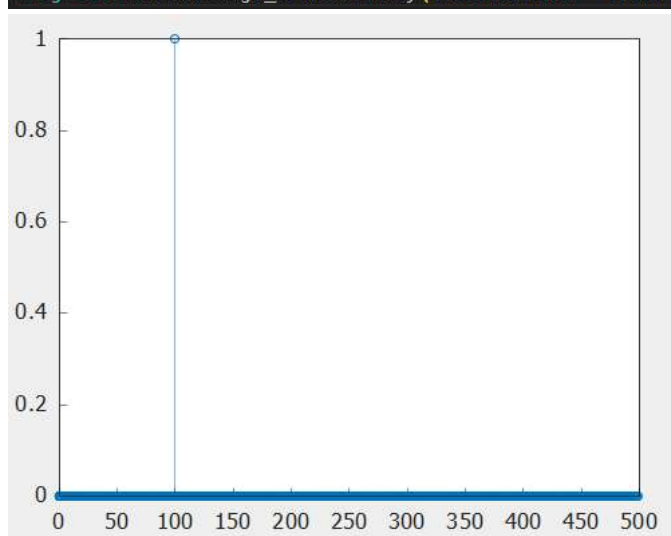
```
sygnał = Projekt3.generuj_sin(czestotliwosc=100, czestotliwosc_probek=1000, czas_trwania=1.0)  
Projekt3.wizualizacja_sygnalu(sygnał = sygnał)
```



Następnie tworzymy z niego transformatę (Jako oś x przyjęliśmy częstotliwość a jako oś y przyjmujemy amplitudę transformaty).

- Generacja Transformaty.

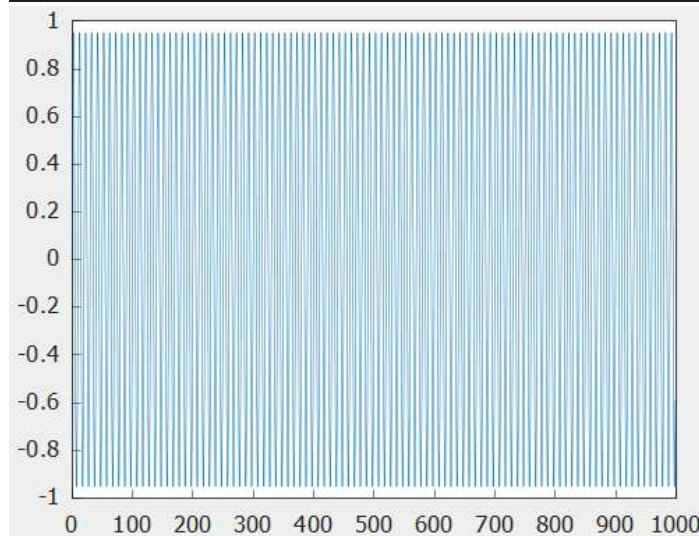
```
transformata = Projekt3.transformata(sygnał = sygnał)  
Projekt3.wizualizacja_transformaty(transformata = transformata, czestotliwosc_probek = 1000)
```



Dla sprawdzenia wygenerujemy transformatę odwrotną żeby sprawdzić czy wyjdzie sygnał który został wygenerowany na początku.

- Transformata Odwrotna.

```
otransformata = Projekt3.odwrotna_transformata(transformata = transformata)
Projekt3.wizualizacja_sygnalu(sygnal = otransformata)
```



Jak widać wyszedł ten sam sygnał czyli transformata działa poprawnie.

### 3) Filtracja 1D

Zastosujmy Filtrację 1D dla przykładowego sygnału i dla przykładowego filtra.

- Przykładowy Filtr i Sygnał.

```
sygnał4 = [1, 2, 3]
filtr = [1, 0, -1]
```

- Wynik powstały w procesie filtracji.

```
wynik = Projekt3.filtracja_1D(sygnał4, filtr)
print(wynik)
```

```
[1.0, 2.0, 2.0, -2.0, -3.0]
```