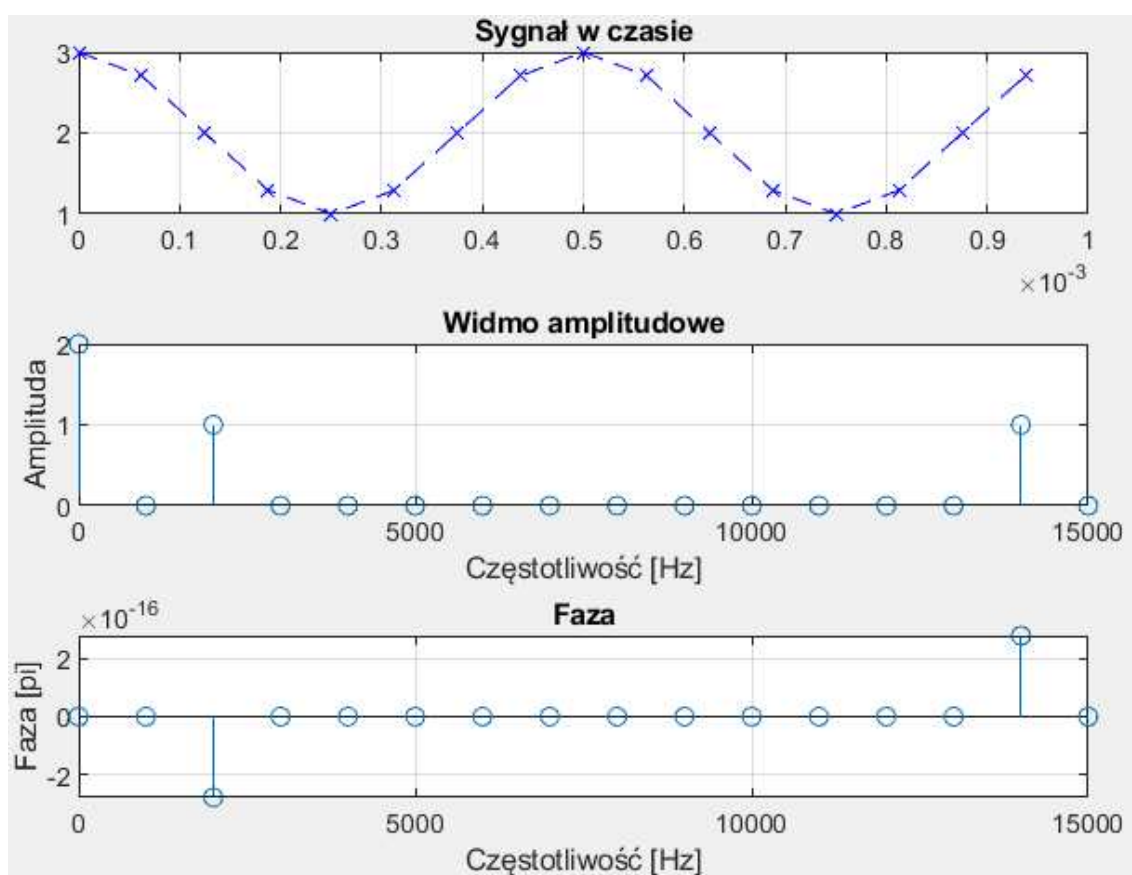
	<p style="text-align: center;"><b>Akademia Górniczo-Hutnicza w Krakowie Wydział FIIS Fizyka Techniczna</b></p>	<p><b>Zrealizował:</b> Ryś Przemysław</p>
Rok akademicki: 2022/2023	Semestr VI	Grupa lab. 2
<p>Temat ćwiczenia:</p> <p style="text-align: center;">5-7.Dyskretna transformata Fourier'a, kompresja JPG</p>		
<p>Data wykonania ćwiczenia</p> <p style="text-align: center;">22.05.2023</p>	<p>Data oddania sprawozdania</p> <p style="text-align: center;">22.05.2023</p>	<p style="text-align: center;">Ocena</p>

## 1. Podstawy DFT.

### Wprowadzenie do DFT.



Rys. 1: wykresy sygnał, widmo oraz faza dla przypadku sinusa ze składową stałą

Wzór na normowanie amplitudy:

$$amplitudeNorm = \frac{amplitude}{\frac{N}{2}} \quad (1)$$

Wzór na częstotliwość bazową:

$$Fbase = \frac{Fsmp}{N} \quad (2)$$

W naszym przypadku wartość  $Fsmp$  wynosi 16000 [Hz], a  $N$  wynosi 16 próbek. Wstawiając te wartości do wzoru, otrzymamy:

$$Fbase = \frac{16000}{16} = 1000 \text{ [Hz]} \quad (3)$$

korzystając z funkcji  $phase = \angle(spectrum) ./ \pi$  otrzymamy fazy prążków znormalizowane do  $\pi$ . W sensie wykonanej normalizacji prążki sinusa odpowiadają fazie 0 lub 1, natomiast cosinusa -0.5 lub 0.5.

Faza prążków sinusa (w $\pi$ ):							
1	2	3	4	5	6	7	8
0	0	1	0	0	0	0	0

Faza prążków kosinusa (w $\pi$ ):							
9	10	11	12	13	14	15	16
0	0	0	0	0	0	0	0

Rys. 2: Fazy prążków sinusa i cosinusa zredukowane do liczby  $\pi$ .

- Faza prążków sinusa (indeksy 1-8): Wszystkie wartości fazowe wynoszą 0, z wyjątkiem indeksu 3, dla którego wartość fazowa wynosi 1 (równa  $\pi$ ). Oznacza to, że prążek sinusa o indeksie 3 ma fazę równą  $\pi$ , podczas gdy pozostałe prążki sinusa mają fazę równą 0.
- Faza prążków cosinusa (indeksy 9-16): Wszystkie wartości fazowe wynoszą 0. Oznacza to, że wszystkie prążki kosinusa mają fazę równą 0.

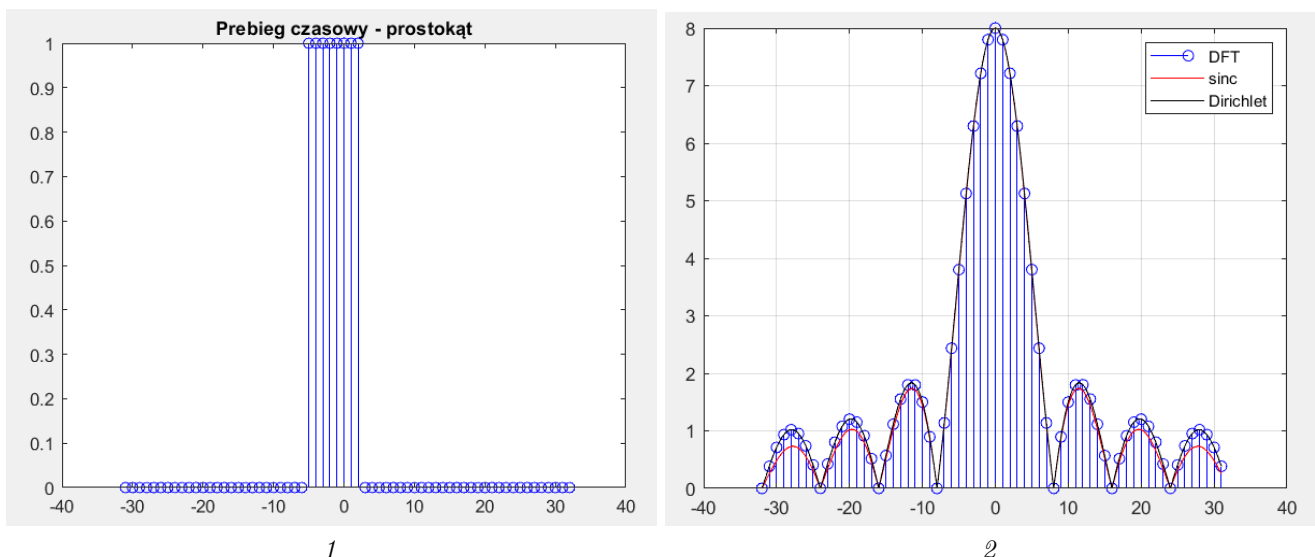
W przypadku podanego sygnału, którego częstotliwość wynosi 2000 Hz, amplituda wynosi 1, faza wynosi  $0.5\pi$  (czyli 90 stopni), a składowa stała wynosi 2, fazy prążków sinusa i kosinusa wynoszą 0, ponieważ odpowiadają one składowym o częstotliwości bazowej (czyli 0 Hz). Składowe o częstotliwości bazowej nie mają fazowej zmienności, więc ich faza wynosi 0.

Widmo amplitudowe pokazuje, że na częstotliwości bazowej (0 Hz) istnieje składowa stała, reprezentowana przez pierwszy prążek widma. Składowa stała nie ma składowej sinusoidalnej, dlatego faza tego prążka wynosi 0.

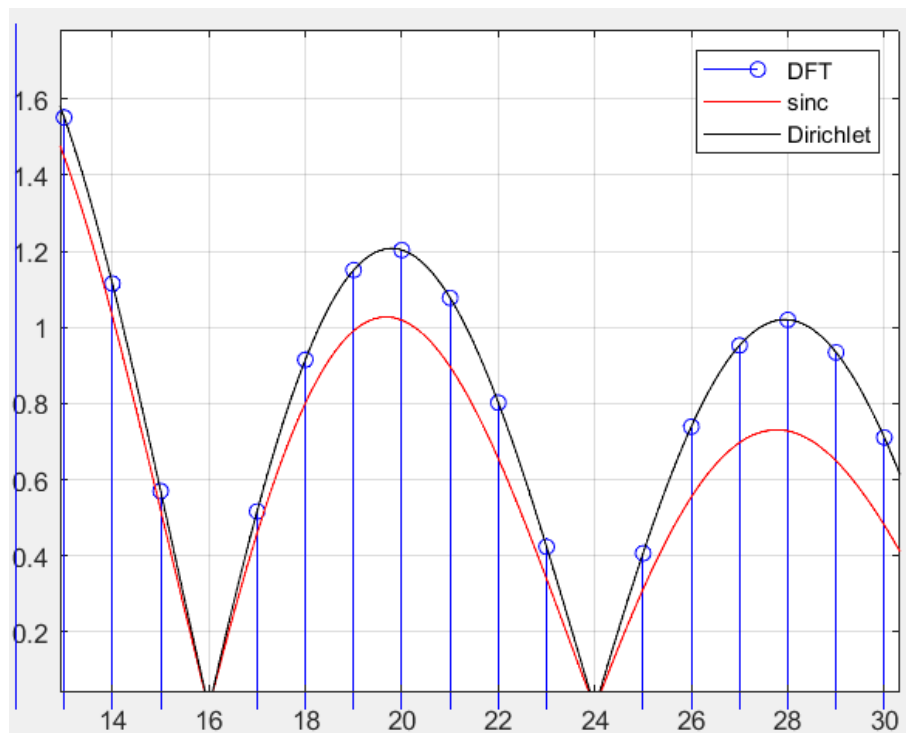
W przypadku sygnałów sinusoidalnych o innych częstotliwościach niż bazowa, prążki sinusa i kosinusa będą miały różne fazy, zgodnie z interpretacją teoretyczną, gdzie prążki sinusa odpowiadają fazie 0 lub  $\pi$ , a prążki kosinusa odpowiadają fazie  $-0.5$  lub  $0.5$ .

## 2. Widmo sygnału prostokątnego.

### Widmo impulsu prostokątnego.



Rys. 3: Widmo impulsu prostokątnego (1) oraz jego DFT (2) dla przypadku  $K = 8$  i  $n_0 = -5$ .



Rys. 4: Zbliżony wykres sygnału prostokątnego po DFT.

Dyskretna transformacja Fouriera (DFT) jest matematycznym narzędziem używanym do analizy sygnałów w dziedzinie częstotliwości. DFT przekształca sygnał dyskretny z dziedziny czasu na dziedzinę częstotliwości, wyrażając go jako kombinację składowych sinusoidalnych o różnych częstotliwościach i amplitudach.

Zgodność wyniku DFT z funkcjami  $\text{sinc}(x)$  i Dirichlet'a wynika z własności matematycznych transformacji Fouriera oraz charakterystyki tych funkcji.

Funkcja  $\text{sinc}(x)$  jest zdefiniowana jako  $\frac{\sin(x)}{x}$  i ma unormowany kształt zbliżony do sinusa, gdzie jej wartość w zerze wynosi 1. Funkcja  $\text{sinc}(x)$  ma również nieskończoną szerokość pasma, co oznacza, że jej widmo rozciąga się na nieskończoność. Widmo  $\text{sinc}(x)$  w dziedzinie częstotliwości jest rozkładem równomiernie rozmieszczonych prążków o amplitudach proporcjonalnych do wartości  $\text{sinc}(x)$  w dziedzinie czasu. W przypadku DFT, jeśli sygnał wejściowy jest odpowiednio próbkowany, to amplitudy prążków w widmie DFT będą proporcjonalne do wartości funkcji  $\text{sinc}(x)$  w dziedzinie czasu. Dlatego w przypadku harmonicznych sygnałów sinusoidalnych, widmo DFT będzie miało wyraźne prążki o amplitudach proporcjonalnych do wartości  $\text{sinc}(x)$  w dziedzinie czasu.

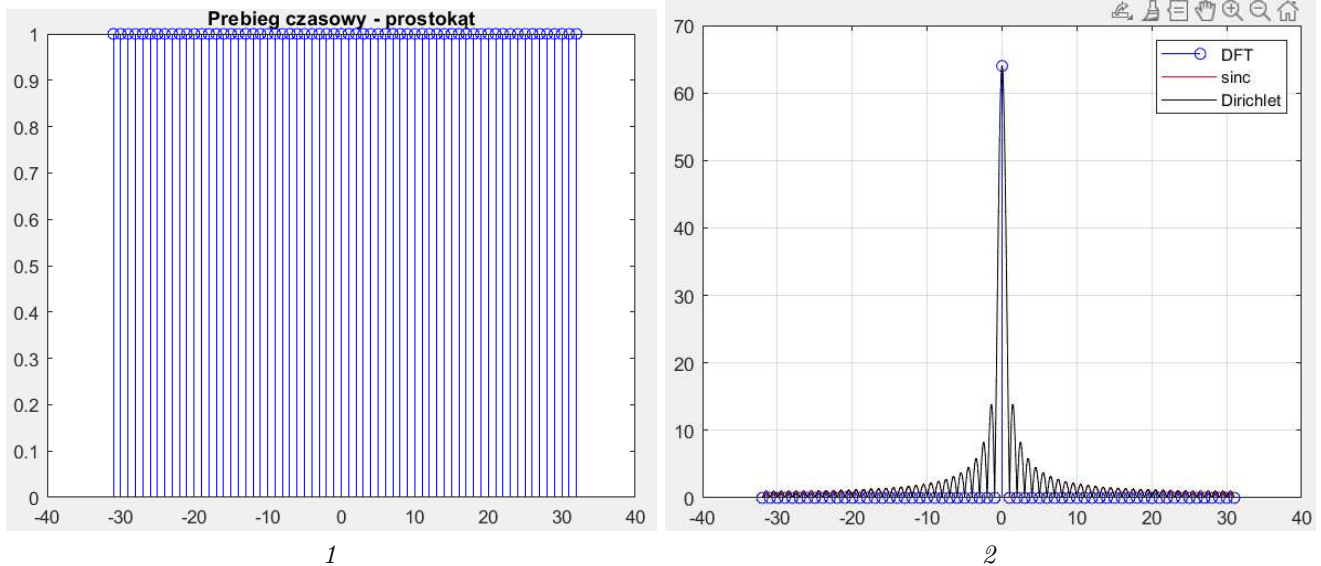
Funkcja Dirichlet'a ( $D_N(x)$ ) jest zdefiniowana jako suma sinusów o różnych częstotliwościach i wagach, co pozwala na reprezentację sygnałów okresowych. Funkcja Dirichlet'a ma widmo składające się z dyskretnych prążków o amplitudach proporcjonalnych do wartości funkcji Dirichlet'a w dziedzinie czasu. Widmo DFT próbkowanego sygnału okresowego będzie miało podobną strukturę z dyskretnymi prążkami, których amplitudy będą proporcjonalne do wartości funkcji Dirichlet'a.

Położenie pierwszego minimum powyżej piku głównego znajduje się w punkcie odpowiadającym próbkę o indeksie 8. Linijka  $\text{pozycja\_min} = \text{pozycja\_min} / N * F_{\text{smp}}$ ; służy do przeliczenia pozycji pierwszego minimum na skalę częstotliwości. Po obliczeniu  $\text{pozycja\_min}$  jako różnicy indeksów próbek, wartość ta jest normalizowana przez ilość próbek  $N$  i mnożona przez częstotliwość próbkowania  $F_{\text{smp}}$ . To przekształcenie ma na celu przeliczenie pozycji minimum z dziedziny indeksów próbek na skalę częstotliwości w odpowiednich jednostkach. Zatem dla  $F_{\text{smp}} = 1$  oraz  $N = 64$  otrzymujemy bezwymiarową częstotliwość odpowiadającą pierwszemu minimum równą 0.125.

Zależność widma amplitudowego od pozycji początku impulsu prostokątnego  $n_0$  wynika z charakterystyki impulsu prostokątnego oraz właściwości dyskretnej transformaty Fouriera (DFT). Impuls prostokątny o szerokości  $K$  próbek, który jest przesunięty o  $n_0$  próbek, ma wpływ na widmo amplitudowe sygnału, ponieważ jego położenie i szerokość wpływa na rozmieszczenie składowych częstotliwościowych w widmie. Teoretycznie, widmo amplitudowe impulsu prostokątnego jest opisane funkcją  $\text{sinc}(x)$  w dziedzinie częstotliwości. Funkcja  $\text{sinc}(x)$  ma maksimum w zerze i ma charakterystyczne zera w punktach, gdzie argument jest równy całkowitej wartości. Dla impulsu prostokątnego, widmo amplitudowe ma minimum zerze i występują kolejne zera w punktach, gdzie częstotliwość jest równa całkowitej wartości szerokości impulsu.

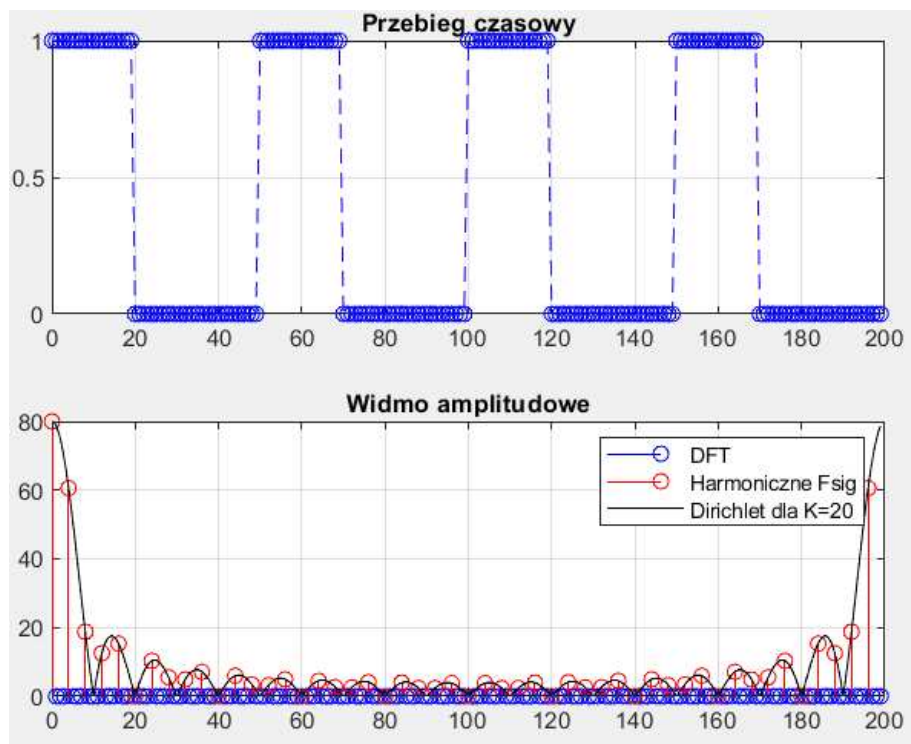
W tym przypadku DFT transformuje sygnał na jeden tylko prążek, który znajduje się w zerze i przykrywa się idealnie zarówno z funkcją Dirichlet'a jak i  $\text{sinc}(x)$ . Funkcje te natomiast rozciągają się dalej co już nie odpowiada transformacji DFT, a ich kolejne zera celują w wartości  $N/K$ . Jest to spowodowane szczególnym przypadkiem sygnału.

Dla przebiegu prostokątnego o nie-50[%] współczynnika wypełnienia (gdzie  $K = 20$ ), widmo amplitudowe będzie miało niezerowe prążki dla harmonicznych o częstotliwościach wielokrotnościach podstawowej częstotliwości sygnału ( $f$ ). Dlatego będziemy obserwować harmoniczne w tych niezerowych prążkach.

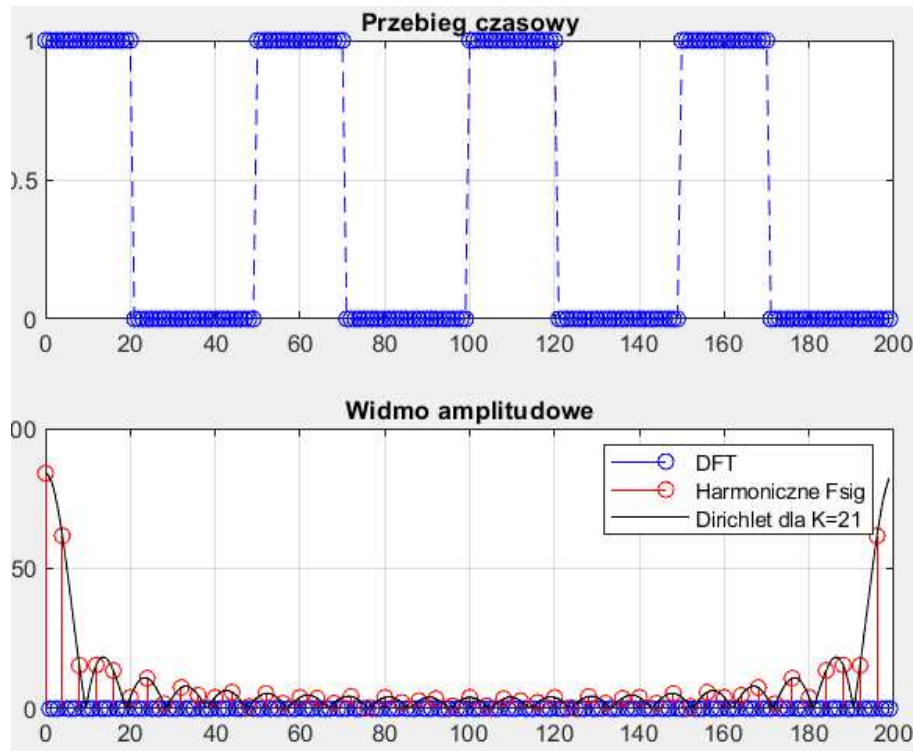


Rys. 5: Widmo impulsu prostokątnego (1) oraz jego DFT (2) dla przypadku  $K = N$  i  $n_0 = -(K-1)/2$ .

### Współczynnik wypełnienia.



Rys. 6: Przebieg czasowy i przebieg amplitudowy dla sygnału prostokątnego dla szerokości dodatniej połówki przebiegu  $K = 20$ .



Rys. 7: Przebieg czasowy i przebieg amplitudowy dla sygnału prostokątnego dla szerokości dodatniej połówki przebiegu  $K = 21$ .

Harmoniczne dla sygnału prostokątnego są wielokrotnościami częstotliwości podstawowej. W przypadku sygnału prostokątnego o okresie  $T$  i szerokości  $K$ , częstotliwość podstawowa (pierwsza harmoniczna) wynosi  $f = 1/T$ . Dla danego  $K$ , harmoniczne będą miały częstotliwość  $f_n = n \cdot f$ , gdzie  $n$  jest liczbą naturalną większą od zera. W naszym przypadku mamy  $K = 21$ , co odpowiada szerokości dodatniej połówki przebiegu. Mając  $T = 50$ , częstotliwość podstawowa  $f = 1/50 = 0.02$ . Teraz możemy obliczyć częstotliwości harmonicznnych dla pierwszych 10 harmonicznnych. Aby przeliczyć częstotliwości na indeksy, należy podzielić częstotliwość przez częstotliwość próbkowania  $F_s$  i pomnożyć przez długość ciągu próbek  $N$ , przyjmując  $F_s = 1$  otrzymujemy:

- $f_1 = f = 0.02 \rightarrow id_1 = f_1 \cdot N/F_s = 0.02 \cdot 200 = 4$
- $f_2 = 2f = 0.04 \rightarrow id_2 = f_2 \cdot N/F_s = 0.04 \cdot 200 = 8$
- $f_3 = 3f = 0.06 \rightarrow id_3 = f_3 \cdot N/F_s = 0.06 \cdot 200 = 12$
- $f_4 = 4f = 0.08 \rightarrow id_4 = f_4 \cdot N/F_s = 0.08 \cdot 200 = 16$
- $f_5 = 5f = 0.1 \rightarrow id_5 = f_5 \cdot N/F_s = 0.1 \cdot 200 = 20$
- $f_6 = 6f = 0.12 \rightarrow id_6 = f_6 \cdot N/F_s = 0.12 \cdot 200 = 24$
- $f_7 = 7f = 0.14 \rightarrow id_7 = f_7 \cdot N/F_s = 0.14 \cdot 200 = 28$
- $f_8 = 8f = 0.16 \rightarrow id_8 = f_8 \cdot N/F_s = 0.16 \cdot 200 = 32$
- $f_9 = 9f = 0.18 \rightarrow id_9 = f_9 \cdot N/F_s = 0.18 \cdot 200 = 36$
- $f_{10} = 10f = 0.2 \rightarrow id_{10} = f_{10} \cdot N/F_s = 0.2 \cdot 200 = 40$

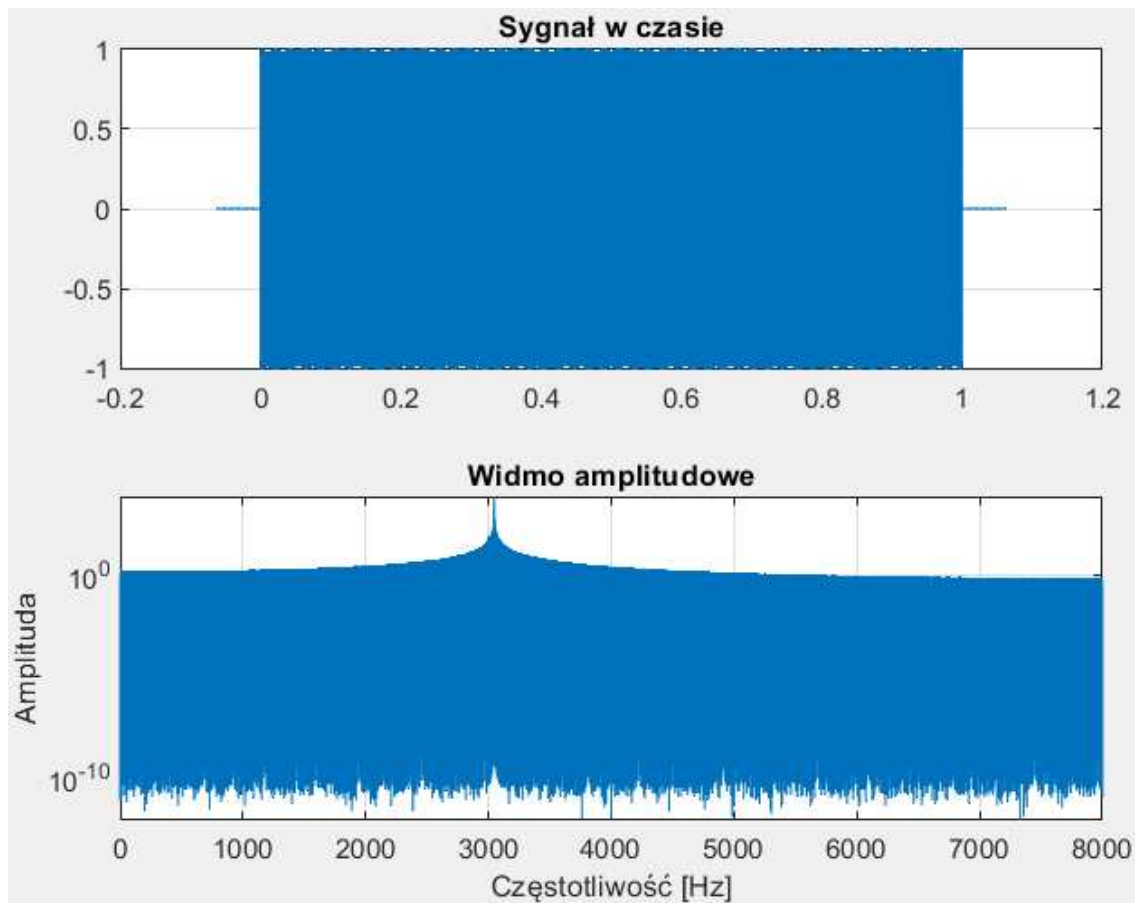
Dla  $K = 20$  indeksy niezerowych prążków odczytane z wykresu to kolejno:  $id_1, id_2, id_3, id_4, id_5, id_7, id_8, id_9, id_{10}$

W podanym zakresie zero wynoszą zatem te o indeksach:  $id_6, id_{11}$

## 6. Rozdzielczość transformaty DFT.

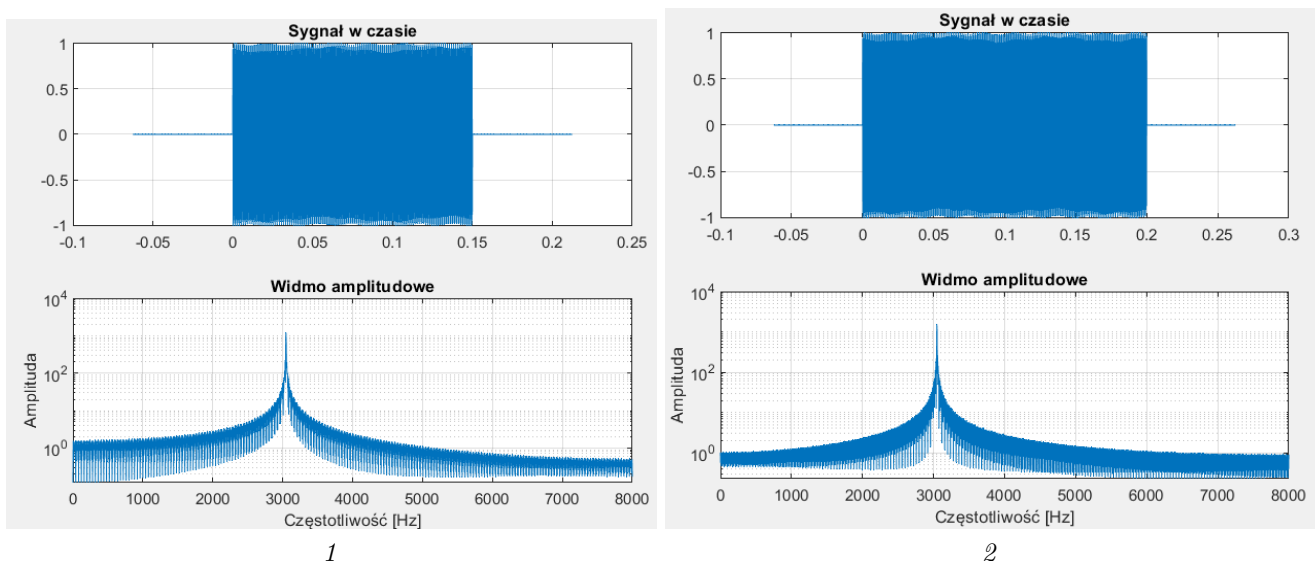
### Rozdzielczość a wyciek.

Dla wartości zeroPadding około 1000 położenie maksimum widma zaczyna odpowiadać zadanej wartości  $F_{sig2} = 3047$  [Hz].



Rys. 8: Widmo sygnału w czasie oraz widmo amplitudowe dla zeroPadding = 1000.

W przypadku analizy sygnałów o nieidealnych okresach lub częstotliwościach, takich jak sygnały o niecałkowitych wielokrotnościach podstawowej częstotliwości, w wyniku może wystąpić efekt wycieku widma. Efekt ten powoduje, że energia sygnału rozprzestrzenia się na sąsiednie prążki widma, utrudniając dokładne zlokalizowanie pików dla konkretnych częstotliwości. Dodanie zer na końcu sygnału przed wykonaniem DFT, czyli zero padding, ma wpływ na poprawę rozdzielczości częstotliwości widma. Poprzez dodanie zer, rozszerzamy czas analizowanego sygnału, co z kolei prowadzi do zwiększenia liczby prążków w wynikowym widmie DFT. Dodatkowe zera w sygnale pozwolą na lepsze wychwycenie i zlokalizowanie energii sygnału w widmie. Dzięki temu, jeśli istnieją składniki harmoniczne o nieidealnych częstotliwościach, takie jak częstotliwości niebędące dokładnymi wielokrotnościami częstotliwości próbkowania, zero padding umożliwia lepsze wyodrębnienie ich energii i dokładniejsze zlokalizowanie ich w widmie DFT. Jednak warto pamiętać, że zero padding nie zwiększa rzeczywistej rozdzielczości częstotliwości, ale jedynie wpływa na dokładniejsze zlokalizowanie istniejących składników harmonicznych.



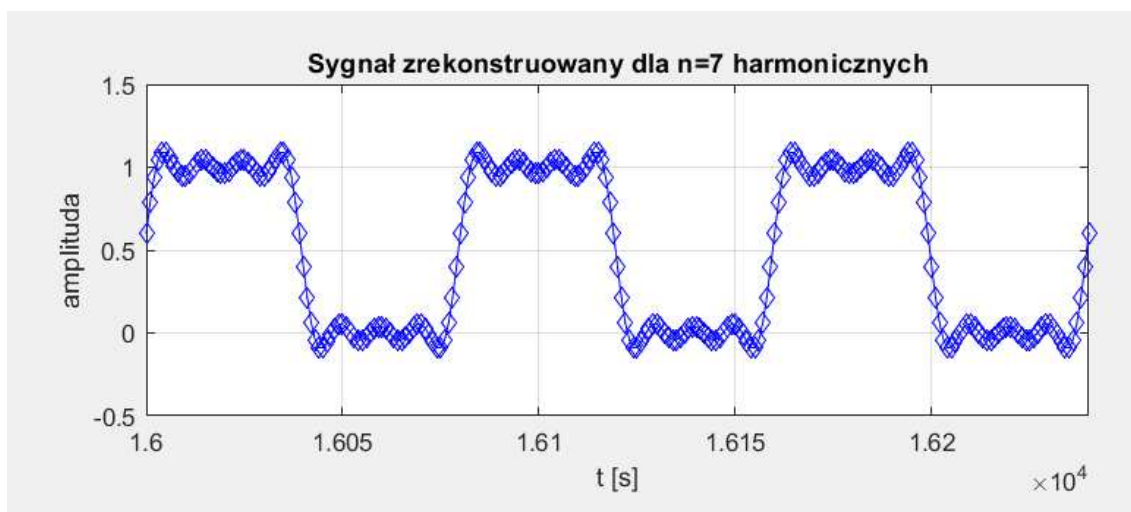
Rys. 9: Widmo sygnału oraz amplitudowe dla  $F_{\text{sig1}} = 3000$  [Hz] i  $N = 2400$  (1) oraz 3200 (2).



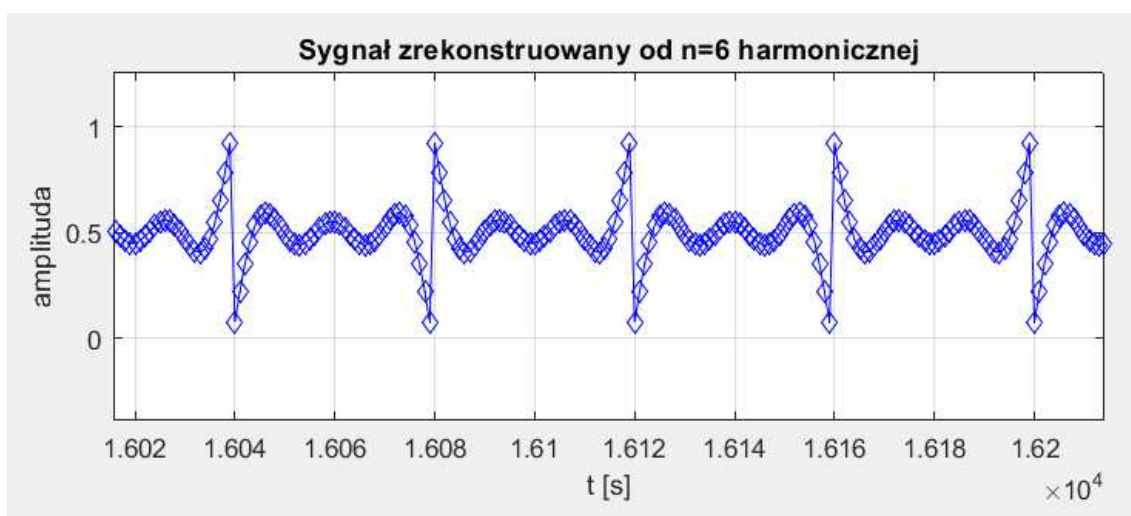
Przyglądając się wykresowi widma amplitudowego dla  $N = 2400$  i  $N = 3200$ , zauważamy, że dla  $N = 2400$  pik o częstotliwości 3 kHz jest dobrze widoczny jako wyraźny prążek, podczas gdy dla  $N = 3200$  pik ten staje się mniej widoczny i trudniejszy do rozróżnienia od sąsiednich prążków. Zwiększanie  $N$  (liczby próbek) ogólnie powinno pomóc w lepszym wyodrębnieniu częstotliwości sygnałów w widmie. Większa liczba próbek oznacza większą rozdzielczość czasową, co przekłada się na lepszą rozdzielczość częstotliwościową w dziedzinie częstotliwości. To pozwala na dokładniejsze rozdzielenie blisko położonych pików spektralnych. Dlaczego jednak zwiększenie  $N$  do 3200 nie poprawiło widoczności pików o częstotliwości 3 kHz? W przypadku  $N = 3200$ , częstotliwość próbkowania ( $F_{\text{smp}}$ ) jest stała, ale długość sygnału ( $N$ ) została zwiększona. To oznacza, że długość okna czasowego (okres, na którym obliczane jest FFT) wzrosła, co skutkuje zmniejszeniem rozdzielczości częstotliwościowej. Mimo większej liczby próbek, nie uzyskujemy większej precyzji w lokalizacji konkretnych częstotliwości, a sąsiednie prążki spektralne mogą się na siebie nachodzić. W rezultacie pik o częstotliwości 3 kHz staje się mniej widoczny, a odróżnienie go od sąsiednich prążków staje się trudniejsze. W celu poprawy widoczności pików o częstotliwości 3 kHz, warto zwiększyć zarówno liczbę próbek ( $N$ ) jak i częstotliwość próbkowania ( $F_{\text{smp}}$ ), aby uzyskać większą rozdzielczość częstotliwościową i lepszą lokalizację pików spektralnych.

## 7. Transformata odwrotna – filtracja w dziedzinie częstotliwości.

### Filtracja sygnału prostokątnego.



Rys. 10: Wykres sygnału zrekonstruowanego dla pierwszego filtru.



Rys. 11: Wykres sygnału zrekonstruowanego dla drugiego filtru.

Istnieje wiele różnych filtrów, które można zastosować w celu poprawy jakości i kształtu zrekonstruowanego przebiegu sygnału, w programie dokonujemy teoretycznej implementacji następujących filtrów:

1. Filtr dolnoprzepustowy (low-pass filter): Ten rodzaj filtru przepuszcza składowe o niskich częstotliwościach i tłumi składowe o wysokich częstotliwościach. Stosowany jest w celu eliminacji szumów wysokoczęstotliwościowych i

wygładzenia przebiegu sygnału. Może wpływać na kształt zrekonstruowanego przebiegu poprzez wygładzanie ostrych zmian i usunięcie detali o wysokiej częstotliwości.

Ten filtr w mniejszej liczbie kroków rekonstrukcji lepiej oddaje prawdziwy obraz zrekonstruowanego sygnału.

2. Filtr górnoprzepustowy (high-pass filter): Ten rodzaj filtru przepuszcza składowe o wysokich częstotliwościach i tłumi składowe o niskich częstotliwościach. Jest używany do usuwania niskoczęstotliwościowego szumu i odrzucania składowych stałych. Filtr górnoprzepustowy może wpływać na kształt zrekonstruowanego przebiegu poprzez wzmocnienie lub podkreślenie szybkich zmian i detali o wysokiej częstotliwości.

Ten filtr efektywnie potrzebuje większej liczby kroków rekonstrukcji, na rysunku 11 niemniej widać do czego ta rekonstrukcja zbiega.

Wpływ filtrów na kształt zrekonstruowanego przebiegu sygnału zależy od wielu czynników, takich jak charakterystyka filtru, zakres częstotliwości, stopień tłumienia, a także zawartość widmowa sygnału.



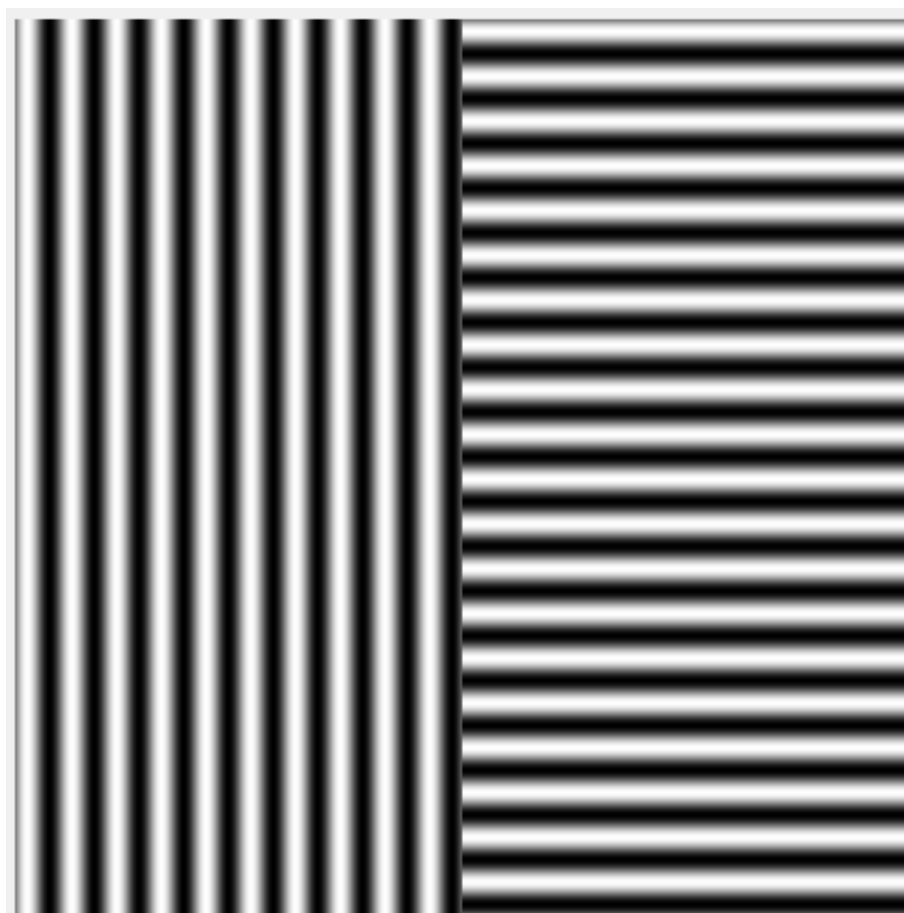
Rys. 12: Wykres w pełni zrekonstruowanego sygnału.

Dla obu przypadków wynik rekonstrukcji dąży do tego samego co przedstawione jest na rysunku 12.

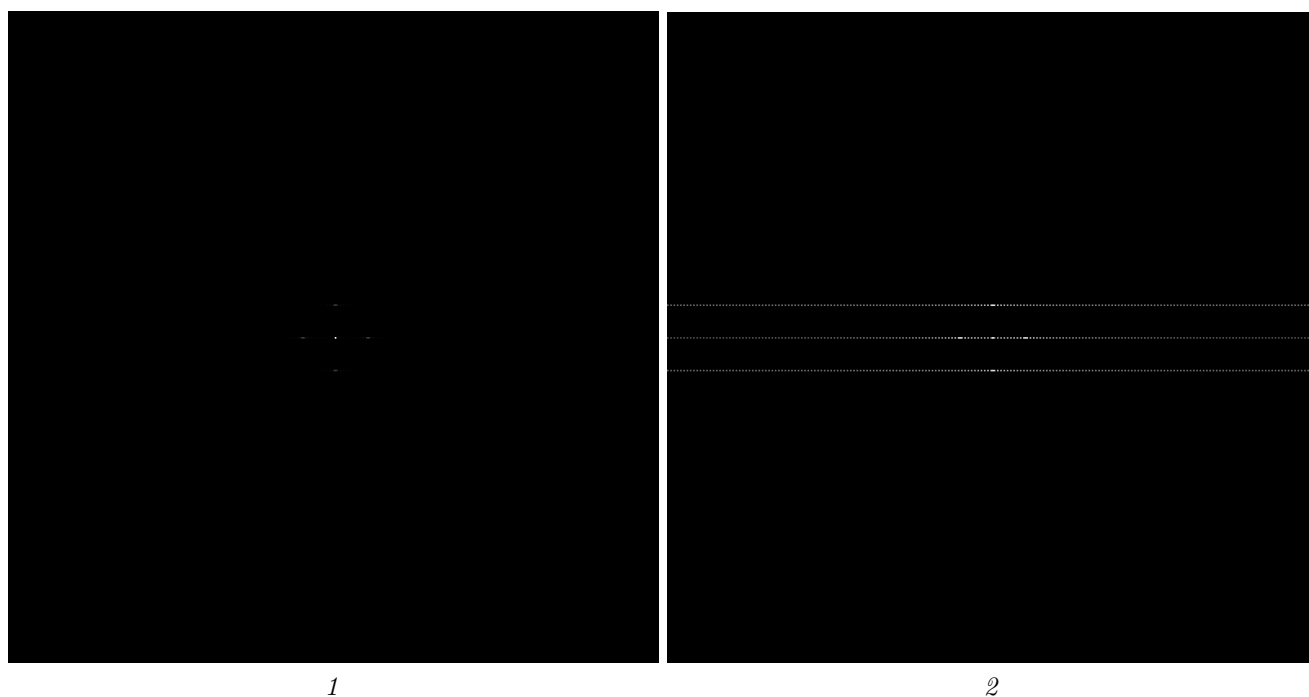


## 8. Transformata DFT obrazu.

Podstawy DFT obrazu – obrazy bardziej złożone.



Rys. 13: Obrazek wzięty w celu transformacji DFT.



Rys. 14: Widmo transformacji DFT w skali liniowej (1) oraz logarytmicznej (2).

Widmo "Dyskretnej Transformaty Fouriera"(DFT) obrazka reprezentuje rozkład składowych częstotliwości obec-

nych w obrazie. Składowe o niższych częstotliwościach są zazwyczaj odpowiedzialne za ogólny kształt i strukturę obrazu, podczas gdy składowe o wyższych częstotliwościach reprezentują bardziej szczegółowe elementy, takie jak krawędzie i tekstury.

Wartości amplitudowe w widmie DFT odzwierciedlają intensywność poszczególnych składowych częstotliwości. Wyższe amplitudy wskazują na większą obecność danej częstotliwości w obrazie.

Widmo DFT obrazka jest zazwyczaj symetryczne wokół środka obrazka. Lewa połowa widma odzwierciedla składowe częstotliwości dla położenia poziomego, podczas gdy prawa połowa odzwierciedla składowe częstotliwości dla położenia pionowego. Symetria ta wynika z faktu, że DFT obrazka jest zwykle obliczana na dwuwymiarowej macierzy danych. Często widmo DFT jest wyświetlane w skali logarytmicznej, co pozwala na lepszą wizualizację szerokiego zakresu wartości amplitudowych. Skala logarytmiczna pozwala uwidocznić zarówno niskie, jak i wysokie amplitudy.

Widmo DFT może również zawierać pewne artefakty wynikające z dyskretnego charakteru DFT i sposobu, w jaki obraz jest poddawany transformacji. Na przykład, artefakty mogą się pojawić w postaci tzw. efektu "lejka" (ang. "ringing effect"), który jest wynikiem skończonej liczby próbek obrazu.

## 9. Filtracja zdjęć w dziedzinie częstotliwości.

### Filtracja dolnoprzepustowa zdjęć.



Rys. 15: Obraz przed zastosowaniem filtru.



Rys. 16: Obraz po zastosowaniu filtra typu koło.



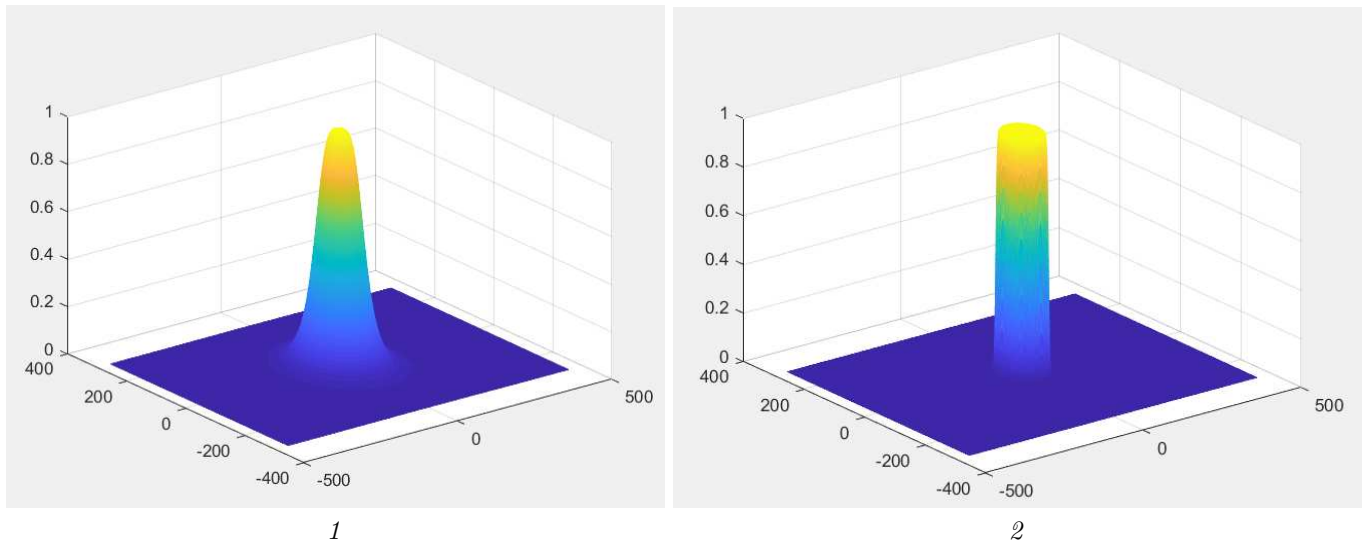
Rys. 17: Obraz po zastosowaniu filtra typu Butterworth-a rzędu 2.

W kontekście do zadania 8.4 jak wiemy filtr dolnoprzepustowy działa poprzez rozmycie obrazu, ograniczając wysokie częstotliwości. Działa on jako "maska", która eliminuje bardzo subtelne szczegóły i ostre krawędzie, które mogą przyczyniać się do występowania efektu Moiré. Moiré to niepożądane zjawisko, które może powodować zakłócenia i zniekształcenia obrazu, tworząc charakterystyczne, faliste wzory. W wyniku zastosowania filtra dolnoprzepustowego obraz może wydawać się nieco mniej ostry, ale jednocześnie zminimalizowane zostają zakłócenia związane z Moiré.



Rys. 18: Zdjęcie z zaznaczonym obszarem na którym w znaczny sposób widać efekt Moiré.

Na rysunku 18 zauważalne są zafalowania związane z efektem Moiré. Dolnoprzepustowy filtr ma za zadanie wygładzić obraz, redukując wysokie częstotliwości, czyli detale i zmiany pikseli o wysokich wartościach. Pozwala to na utrzymanie lub podkreślenie niskich częstotliwości, które reprezentują ogólne struktury obrazu. Efekt filtru dolnoprzepustowego typu koło na zdjęciu będzie polegał na zmniejszeniu ostrości krawędzi i szczegółów, a jednocześnie na utrzymaniu ogólnych struktur i obszarów o płynnych zmianach. Obraz może wyglądać bardziej rozmycie i mniej szczegółowo, ale jednocześnie może zyskać na płynności i jednolitości. Lepsze efekty rozmycia uzyskiwane są w wyniku zastosowania filtru Butterworth-a z powodu na "płynniejsze" przejście między odcinanymi częstotliwościami. Z tego też powodu efektu nie da się zauważyć dla filtru Butterworth-a niższego rzędu.



Rys. 19: Filtr Butterwortha rzędu 2 (1) oraz 20 (2).

Filtr typu koło powoduje ostre odcięcie, natomiast, filtr Butterworth-a działa w sposób przedstawiony na rysunku 19. Można zauważyć, iż w miarę wzrostu rzędu filtru jego "koło odcięcia" staje się ostrzejsze.

### Prosty algorytm wyostrzający.

Parametry, które moim zdaniem dają najlepsze efekty dla wyostrenia obrazu są następujące:

- filterRatio = 0.2



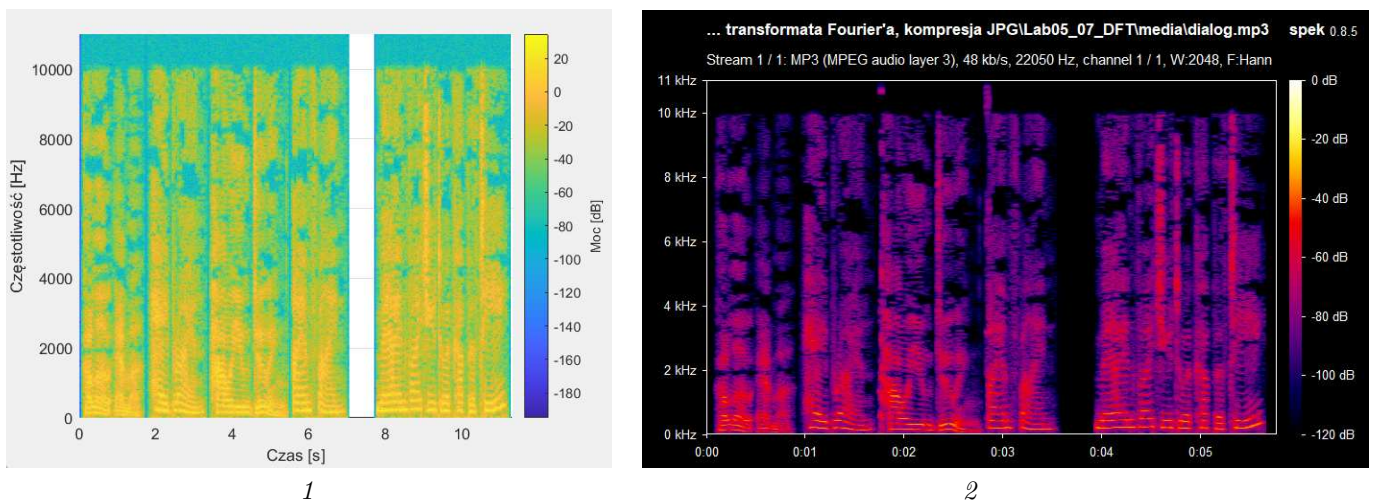
- `butterFilterOrder = 10`
- `sharpeningCoef = 0.4`



Rys. 20: Efekt zastosowania powyższych parametrów wyostrenienia przed (1) oraz po (2).

## 10. Kompresja dźwięku i obrazu.

### MDCT – spektrogram.



Rys. 21: Spektrogram dla pliku dialog.mp3 z wykorzystaniem programu Matlab (1) oraz Spek (2).

Długość ramki MDCT ma bezpośredni wpływ na rozdzielczość częstotliwościową spektrogramu. Im dłuższa jest ramka, tym lepsza jest rozdzielczość w dziedzinie częstotliwości. Dłuższa ramka oznacza większą liczbę próbek, które podlegają transformacji. W wyniku tego mamy większą liczbę prążków w wynikowym spektrum. Większa liczba prążków oznacza większą precyzję w określaniu częstotliwości składowych dźwięku. Zwiększona rozdzielczość częstotliwościowa pozwala na dokładniejsze wykrycie subtelnych zmian w spektrum dźwięku, takich jak harmoniczne, formanty czy szumy.

Zwiększenie długości ramki MDCT ma natomiast odwrotny wpływ na rozdzielczość czasową spektrogramu. Im dłuższa jest ramka, tym gorsza jest rozdzielczość w dziedzinie czasu. Dłuższa ramka oznacza, że analizowany fragment dźwięku obejmuje większy czas. To prowadzi do większego rozmycia informacji czasowych w spektrogramie. Jeśli występują szybkie zmiany w sygnale dźwiękowym, takie jak transjenty czy krótkotrwałe zdarzenia, mogą one zostać zatarte w wyniku dłuższej ramki. W praktyce, jeśli zależy nam na analizie zjawisk krótkotrwałych w czasie, preferujemy krótsze ramki MDCT, które zapewniają lepszą rozdzielczość czasową. Dlatego, wybór długości ramki MDCT to kompromis między rozdzielczością częstotliwościową a czasową. Dłuższa ramka zapewnia lepszą rozdzielczość częstotliwościową, ale gorszą rozdzielczość czasową, podczas gdy krótsza ramka ma odwrotne właściwości. Ostateczny wybór zależy od konkretnego przypadku zastosowania i preferencji analizującego.

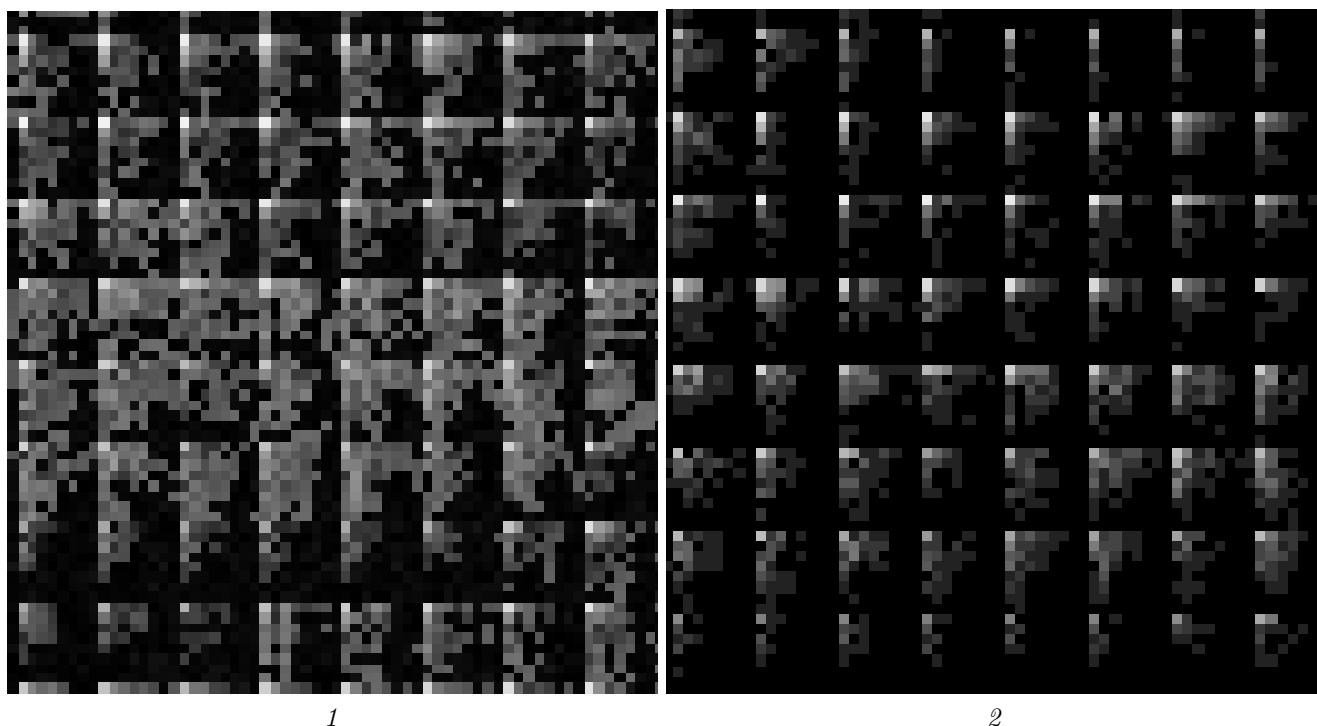
## Kompresja obrazu – JPEG.

Wartość parametru quality dającego jeszcze akceptowalny rezultat wynosi: quality = 65 [%]. Uzyskany stopień kompresji DCT wynosi 99.34 [%], a współczynnik kompresji JPEG wynosi 29.23 [%].



Rys. 22: Oryginalne zdjęcie sprzed kompresji (1) oraz po dla wyżej zadanej wartości quality (2).

Z porównania obrazów 1 i 2 z rysunku 22 można zauważyć, iż po kompresji obraz stracił na głębi, cienie wyblakły, a barwy stonowały się w bardziej monotonne odcienie szarości.



Rys. 23: Przedstawienie wartości wybranego bloku 8x8 po transformacji DCT i jej kwantyzacji.

W przypadku kompresji JPEG, algorytm DCT stosuje się do bloków pikseli obrazu. Bloki te są przekształcane z dziedziny przestrzennej do dziedziny współczynników DCT. W wyniku tego przekształcenia, obraz jest reprezentowany jako zestaw współczynników DCT, które reprezentują różne składowe częstotliwości w danym bloku.

Przekształcenie DCT pozwala na reprezentację obrazu za pomocą mniej informacji, ponieważ często wiele współczynników DCT o niższych częstotliwościach ma mniejsze wartości i może być bardziej efektywnie skwantyzowane, a nawet usunięte w procesie kompresji. W efekcie, większa liczba współczynników DCT o wyższych częstotliwościach, które reprezentują drobne szczegóły obrazu, może zostać zredukowana lub usunięta, co prowadzi do kompresji danych.

Kompresja stratna JPEG redukuje ilość danych poprzez kwantyzację wartości pikseli, co oznacza, że mniejsze wartości są zaokrąglane lub ucięte, co prowadzi do utraty szczegółów i subtelnych zmian w obrazie. Wykorzystanie kodowania entropijnego, aby skompresować dane, eliminując redundancję informacji. Redukcja ilości danych prowadzi do mniejszego rozmiaru pliku ze zdjęciem, co jest korzystne dla przechowywania i przesyłania obrazów w sieci.



Kompresja stratna JPEG ma wpływ na jakość obrazu ze względu na utratę informacji w procesie kompresji. Zwiększenie stopnia kompresji (większe wartości kwantyzacji) prowadzi do większej utraty szczegółów i większej degradacji jakości obrazu. Obrazy skompresowane w formacie JPEG są zazwyczaj mniej ostre, mają mniejszą gamę kolorów i mogą wykazywać blokowe artefakty.

W przybliżeniu, kompresja stratna JPEG można porównać do pewnego rodzaju filtracji. Poprzez kwantyzację, informacje o wysokich częstotliwościach, takich jak subtelne tekstury i krawędzie, są usuwane lub ograniczane. Można to interpretować jako działanie filtra dolnoprzepustowego, który przepuszcza niskie częstotliwości i redukuje lub usuwa wysokie częstotliwości. W rezultacie, szczegóły obrazu są redukowane, a obraz jest bardziej "wygładzony".

Biorąc pod uwagę powyższy opis działania kompresji JPEG można zauważyć na rysunku 22, iż rzeczywiście krawędzie jak i intensywniej się zmieniające tekstury uległy wygładzeniu, sprawiając tym samym, iż obraz jest bardziej jednolity.

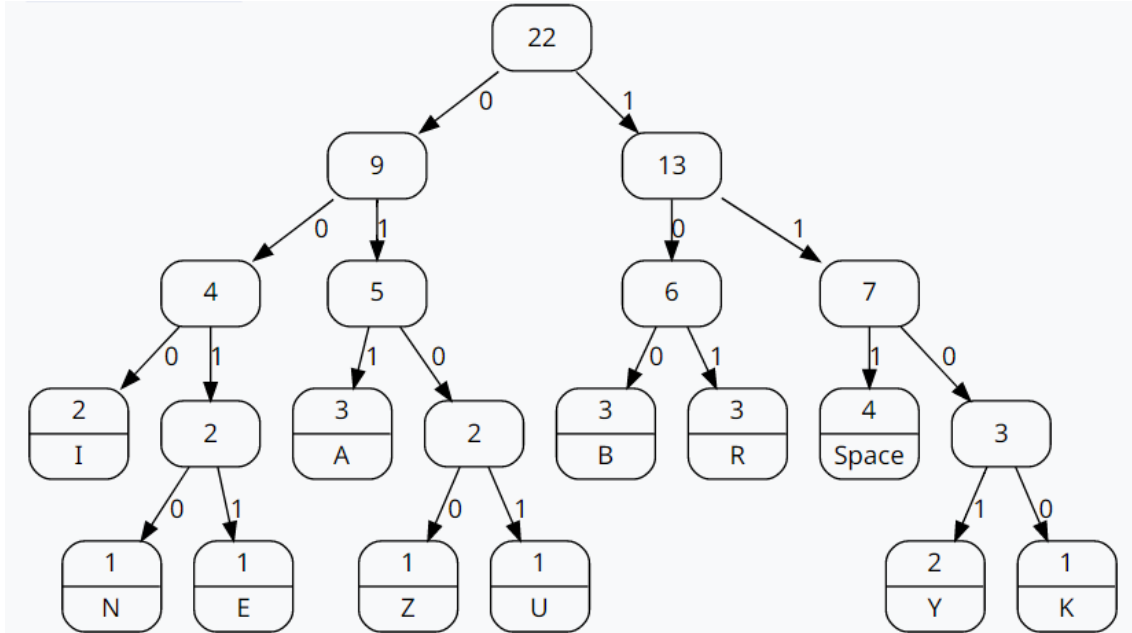
## Kompresja bezstratna – kodowanie Huffman’a.

Analizowany ciąg znaków zwany zdaniem to "NA BEZRYBIU I RAK RYBA". Jego długość liczona w znakach wynosi 22. Liczebności poszczególnych znaków to:

- 'N' występuje 1 raz.
- 'A' występuje 3 razy.
- ' ' występuje 4 razy.
- 'B' występuje 3 razy.
- 'E' występuje 1 raz.
- 'Z' występuje 1 raz.
- 'R' występuje 3 razy.
- 'Y' występuje 2 razy.
- 'I' występuje 2 razy.
- 'U' występuje 1 raz.
- 'K' występuje 1 raz.

Tab. 1: Tabela ilości wystąpień znaków w zadanym zdaniu oraz prawdopodobieństwa ich wystąpienia.

Symbol	Ilość wystąpień	Prawdopodobieństwo
'N'	1	0.045
'A'	3	0.136
' '	4	0.182
'B'	3	0.136
'E'	1	0.045
'Z'	1	0.045
'R'	3	0.136
'Y'	2	0.091
'I'	2	0.091
'U'	1	0.045
'K'	1	0.045



Rys. 24: Drzewo Huffman'a dla wybranego zdania.

Ciąg bitowy reprezentujący zdanie "NA BEZRYBIU I RAK RYBA"wynosi:

$$0010011111100001101001011101100000010111100011110101111001111011101100011 \quad (4)$$

Efektywność algorytmu można przedstawić licząc średnią ilość bitów na znak. Średnią ilość bitów na znak można wyliczyć z samego drzewa wedle formuły:

$$H = \sum_x p(x) \cdot L(x) \quad (5)$$

, gdzie  $L(x)$  to długość (ilość bitów) kodu odpowiadającego symbolowi  $x$ , a  $p(x)$  to prawdopodobieństwo wystąpienia symbolu (znaku)  $x$ . W naszym przypadku będzie to więc:

$$H = 0.045 \cdot 4 + 0.136 \cdot 3 + 0.182 \cdot 3 + 0.136 \cdot 3 + 0.045 \cdot 4 + 0.045 \cdot 4 + 0.136 \cdot 3 + 0.091 \cdot 4 + 0.091 \cdot 3 + 0.045 \cdot 4 + 0.045 \cdot 4 = \underline{\underline{3.307}} \quad (6)$$

Idealną (minimalną możliwą) długość słowa kodowego dla danego ciągu można obliczyć z podstawowego twierdzenia Shannona (Shannon's source coding theorem) jako:

$$H_{ideal} = - \sum_x p(x) \cdot \log_2[p(x)] \quad (7)$$

Dla naszego ciągu dostaniemy więc:

$$H_{ideal} = \underline{\underline{3.247}} \quad (8)$$

Widać więc że kodowanie Huffman'a daje, pomimo relatywnej prostoty algorytmu, niemalże idealnie krótki ciąg kodowy:

$$\frac{3.307}{3.247} \approx 1.0185, \text{ czyli zaledwie } 1.85\% \text{ więcej, niż najkrótszy teoretycznie możliwy.}$$