Otázky k opakování: Úvod do MATLAB

1) Jak se generujte matice A náhodných čísel v intervalu 100 až 1000 o rozměru 5 řádků, 10 sloupců
   `A=900*rand(5,10)+100`

2) Prvky matice `A` v 2. a 5. řádku a zároveň 6. až posledním sloupci nahraďte číslem 0

   `A([2 5],6:end)=0`

3) Spočtěte průměrnou hodnotu `avrA` všech prvků matice `A` nerovnající se nule
   `avrA=mean(A(A~=0))`

4) Vygenerujte řádkový vektor délky 1000 s lineárně klesajícími prvky od 512 do -512
   `linspace(512,-512,1000)`

5) Vygenerujte sloupcový vektor od 1 po 500 s krokem 10 (využijte transpozice)
   `(1:10:500)'`

6) Otočte pořadí prvků ve vektoru B
   `B=B(end:-1:1);`

7) Spočtěte energii signálu ve vektoru `x` (tj. suma kvadrátu všech hodnot)
   `sum(x.^2)`

Otázky k opakování: Signál

8) Z pohledu dělení byste mohli zařadili signál z digitálního teploměru, který odečítá teplotu každých 5 minut:
    a. finitní
    b. infinitní
    c. kauzální
    d. nekauzální
    e. determinovaný - periodický
    f. determinovaný - neperiodický
    g. harmonický
    h. neharmonický
    i. náhodný - stacionární
    j. náhodný - nestacionární
    k. ergodický
    l. neergodický
    m. spojitý v čase
    n. diskrétní v čase
    o. spojitý v amplitudě
    p. kvantizovaný
    q. ekvidistatntně vzorkovaný

Signál je určitě: finitní, kauzální, neergodický, diskrétní v čase, kvantizovaný, ekvidistantně vzorkovaný [a, c, l, n, p, q]

Ale má složku derminovanou (den/noc, roční období), ale i náhodnou-nestacionární (aprílové počasí) [f nebo j]

9) Jaký vzorkovací kmitočet má signál vzorkovaný každou milisekundu?

T_s=0.001 ms $\rightarrow$ $fs=Ts^{-1}$=1/0.0001= 1 000 Hz

10) Signál byl vzorkovaný od času t=0 frekvencí f_s=200 Hz.

    r. Jakému času odpovídá 1 000. vzorek? $t_k= k/f_s$=1000/200=5 s
    s. Kolik bylo zachyceno vzorků v čase 10,65 s? $k= t_k \cdot fs$=10,65*200= 2130

11) Co je Nyquistův–Shannonův vzorkovací teorém?

K zachycení harmonické složky o frekvenci f_{max} musíme průběh vzorkovat více než dvojnásobnou frekvencí:

$f_s > 2f_{max}$

12) Napište alespoň dvě výhody a dvě nevýhody mezi digitálním a analogovým signálem

Digitální:
Datová řada
Strojové zpracování
Snadný převod mezi analogem a digitálem (A/D, D/A)
Zpracování/analýza je numerická
Ukládání a komprese
Odolnost proti rušení

Analog:
Nepřicházíme o informaci
Spojitý čas
Zpracování v reálném čase

13) Jak se v MATLAB generujte a přehraje komorní A (440 Hz) o délce trvání 5s

```
f0=440;
fs=1000;
T=5;
t=linspace(0,T,T*fs);
A=sin(2*pi*f0*t);
sound(A,fs)
```

Otázky k opakování: Fourierova transformace

1) Pro jednotlivé podtypy Fourierovy transformace (FT) uveďte vlastnosti signálu a spektra (spojitost, periodicita):

- Obecná FT: spojitý neperiodický signál, spojité neperiodické spektrum
- Fourierovy řady: spojitý periodický signál, diskrétní neperiodické spektrum
- Diskrétní v čase FT: diskrétní neperiodický signál, spojité periodické spektrum
- Diskrétní FT: diskrétní periodický signál, diskrétní periodické spektrum

2) Uveďte vztahy pro výpočet dopředné a zpětné Fourierovi transformace signálu z(t) a z[n], detailně rozepište úhlový kmitočet omega.

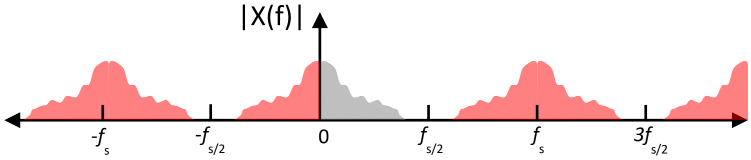
Obecná FT: $Z(f) = \int_{-\infty}^{\infty} z(t) e^{-i\omega t} dt \qquad z(t) = \int_{-\infty}^{\infty} Z(f) e^{i\omega t} df \qquad \omega = 2\pi f$

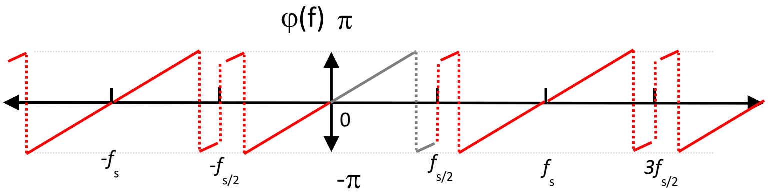
Diskrétní FT: $Z[f_k] = \sum_{n=1}^{N_0} z[n] e^{-i\Omega_k n} \qquad z[n] = \frac{1}{N_0} \sum_{k=0}^{K-1} Z[f_k] e^{i\Omega_k n} \qquad \Omega_k = 2\pi \frac{f_k}{f_s} = 2\pi \frac{k}{N_0}$

3) Jaká bude vzdálenost mezi spektrálními čarami (Δf) signálu délce 1024 vzorků a vzorkovacím kmitočtu 512 Hz.

$\Delta f = f_s/N = 512/1024 = 0.5$ Hz

4) Dokreslete spektrum reálného diskrétního signálu i mimo interval 0-f_s/2





5) Vysvětlete pojem aliasing a jak se mu bránit.
Při nedodržení vzorkovacího teorému jsou všechny frekvenční složky nad fs/2 po navzorkování ozrcadleny (přičteny) pod fs/2. Vzorkovaný signál je tak zkreslen. Řešení: omezení frek. pásma analogovou dolní propustí s mezním kmitočtem <fs/2 před samotným vzorkováním.

6) Jak v MATLAB spočítáte normalizované spektrum signálu `x` se vzorkovacím kmitočtem `fs`? Definujte také vektor `f` obsahující frekvenci odpovídající vektoru spektra `X`. Spočtěte magnitudu `Xm` a fázi `Xphi`.

```
X=(1/length(x))*fft(x);
Xm=abs(X);
Xphi=angle(X);
f=linspace(0,fs-fs/length(X),length(X));
```

Otázky k opakování: Výkonová spektrální hustota

1) Ze signálu o délce N=500 vzorků a vzorkovacím kmitočtu f_s=256 Hz chcete spočítat spektrum $X[f]$ tak, aby spektrální rozlišení bylo Δf=0.5 Hz. Popište postup.

   $\Delta f = \frac{f_s}{M} \rightarrow M = \frac{f_s}{\Delta f} = \frac{256}{0.5} = 512$ vzorků. Spektrum musí být počítáno ze signálu délky 512 vzorků, proto jej doplníme nulami.

   Signál délky N doplníme nulami do délky M tj. za signál přidáme tedy (M-N) 12 nul.

2) Jak vzniká prosakování ve spektru? Jakým způsobem se dá potlačit?

   Nedodržením periodicity signálů, začátek a konec signálu na sebe nenavazují. Použijeme váhovací okno (Hanning, Hamming, apod.)

3) Proč po váhování signálu oknem dochází k rozmazání spektrálních čar?

   Váhovací okno je složeno z více harmonických složek. Násobením signálu s harmonickými složkami okna dochází k heterodynnímu směšování.

4) Jak se změní spektrum signálu doplněný nulami?

   Počet spektrálních čar v intervalu 0-f_s odpovídá počtu vzorků v signále. Dojde tedy k zahuštění spektra (interpolaci spektra). Spektrum neobsahuje více informací oproti původnímu signálu. Je zachována Parsevalova rovnost, proto zahuštěné spektrum má menší magnitudu oproti nezahuštěnému spektru.

5) V MATLAB vytvořte vektor indexů časové segmentace s překryvem. Doba trvání signálu je T=10 s, vzorkovací kmitočet f_s=1.000, segmentační okno w=0.5 s, překryv je okna n=80 %.
   a) jaké bude časové rozlišení (krok mezi okny)? ... sekund, ... vzorků
   b) kolik vznikne segmentačních oken?
   c) v MATLAB vytvořte vektor indexů odkazující na první vzorek segmentačního okna

   $N=T\times f_s=10\times1.000=\underline{10.000}$
   $N_{win}=w\times f_s=0.5\times1.000=\underline{500}$
   $N_{noverlap}=N_{win}\times n=500\times0.8=\underline{400}$
   a) $N_{win}-N_{noverlap}$=500-400=100 vzorků, tj. 100/ f_s=0.1 s (100 ms)
   b) $(N-N_{win})/(N_{win}-N_{noverlap})+1$=9.500/100+1=96 segmentů
   c) `idx=1:Nwin-Nnoverlap:N-Nwin+1;`

6) Popište postup výpočtu odhadu výkonové spektrální hustoty

segment signálu	`seg=signal(idx(i):idx(i)+Nwin-1);`
váhování oknem	`seg=seg.hann(length(seg));`
doplnění nulami (nepovinné)	`seg=[seg(:); zeros(M-N,1)];`
fft	`S=fft(seg);`
kvadrát modulu spektra – tj. PSD	`psd(:,i)=(1/length(seg))*abs(S).^2;`
průměr PSD přes všechny segmenty	`PSD=mean(psd,2);`

7) Signál *x* je vzorkován f_s=1024 Hz. Zvolte segmentační parametry (okno, překryv), aby spektrální rozlišení bylo lepší než Δf=4 Hz a časové rozlišení Δt=125 ms.

$\Delta f < \frac{f_s}{M} \rightarrow M > \frac{f_s}{\Delta f} > \frac{1024}{4} > 256$ vzorků (tj. segmentační okno musí být delší než 0,25 s)

$\Delta t < \frac{N_{win} - N_{noverlap}}{f_s} \rightarrow N_{noverlap} > N_{win} - \Delta t \cdot f_s > 128$ vzorků (tj. $N_{noverlap}/N_{win}$>50%)

Otázky k opakování: Korelace a LTI

1) Napište vzorec vychýlené i nevychýlené křížové korelace, vysvětlete princip a účel korelace.

Vychýlená:

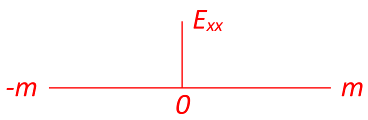
$$Rxy[m] = \sum_{n=1}^{N} x[n+m] \cdot y[n]^*$$

Nevychýlená:

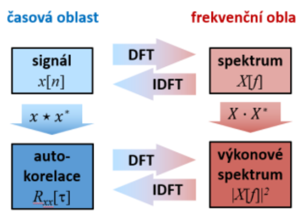
$$Rxy[m] = \frac{1}{N}\sum_{n=1}^{N} x[n+m] \cdot y[n]^*$$

Autokorelace reprezentuje vzájemnou „energii“ (s ponechaným znamínkem) dvojice signálů v závislosti na vzájemném časovém posunu *m* mezi signály. Korelací lze zjistit zpoždění mezi signály a podobnost dvou signálů.

2) Jak vypadá autokorelační funkce šumového signálu. Nakreslete graf a popište osy.

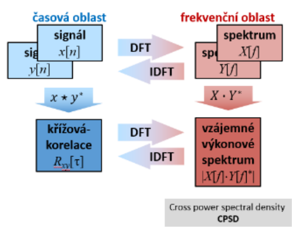


3) Nakreslete a popište vztahy uvnitř DSP čtyřúhelníku pro signál *x*[*n*]



4) Co je to vzájemná výkonová spektrální hustota? Jak ji lze spočítat?

Vzájemná spektrální hustota CPSD je spektrum dvojice signálů *x* a *y* vypočtené z křížové korelační funkce R_{xy}[m] nebo jako skalární násobení spekter signálů **X**·**Y***. CPSD zobrazuje spektrální složky společné pro oba signály.



5) Jak je korelační koeficient, jakých nabývá hodnot a kde se používá?
Korelační koeficient je korelační funkce, kde jednotlivé signály jsou nejprve normalizovány na jednotkovou energii. Korelační koeficient nabývá hodnot od -1 do 1 (-100 % až 100 %) a ukazuje míru podobnosti/inverze dvojice časových řad. Ve statistice se uvádí pro signály bez posunu τ=0.

$$\widehat{Rxy}[\tau] = \frac{Rxy[\tau]}{\sqrt{Rxx[0] \cdot Ryy[0]}}$$

Otázky k opakování: LTI

1) Vysvětlete pojmy:
   a) Kauzalita (na vstup systému navazuje výstup)
   b) Linearita systému (platí princip superpozice)
   c) Časová invariance (systém je v čase neměnný)

2) Co je impulzní odezva systému? Popište v časové i frekvenční oblasti, využijte věty o konvoluci a Fourierově transformaci

   Impulzní odezva je výstup systému po průchodu jednotkového impulzu. Jednotkový impulz, tj. diskrétní nulový signál s jedním jedničkovým vzorkem, jehož spektrum je rovnoměrné. Spektrum impulzní odezvy odpovídá přenosové charakteristice systému. Konvoluce v čase odpovídá násobením ve spektru, proto *Y*[f]=*h*[f] x 1.

3) Spočítejte cyklickou konvoluci x[n] a y[n], kde
   x[n]= 1 2 3 0 -1 -2 0 5 -5
   y[n]= -1 0 1
   x[n]*y[n]= -2 2 4 2 -1 -7 5 4 -7

4) Jaký je rozdíl mezi impulzní odezvou a odezvou na jednotkový skok?

   Impulzní odezva je odpověď systému na jednotkový impulz, jednotkový skok je odpovědí na skokovou změnu vstupu z 0 na 1. Derivací/diferencí odpovědi na jednotkový skok získáme impulzní odezvu.

Otázky k opakování: FIR

1) Co znamená zkratka FIR? Jak souvisí řád filtru, koeficienty filtru a délka impulzní odezvy

   FIR – finite impulse response (konečná impulzní odezva)

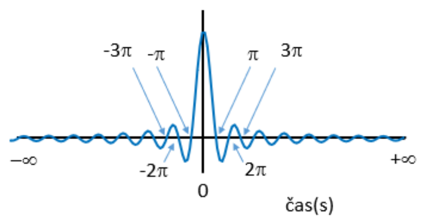
   Po průchodu jednotkového impulzu LTI systémem se na jeho výstupu objeví impulzní odezva konečné délky M+1, kde M je řád filtru. Koeficienty filtru odpovídají impulzní odezvě, tj. konvoluční masce.

2) Označte správné tvrzení o FIR filtrech
   a) Struktura filtru obsahuje pouze dopředné vazby
   b) Struktura filtru obsahuje pouze zpětné vazby
   c) Struktura může obsahovat dopředné i zpětné vazby
   d) Lineární zpoždění je v případě symetrické/antisymetrické impulzní odezvy
   e) Lineární zpoždění je v případě nesymetrické impulzní odezvy
   f) Zpoždění se kompenzuje posunem výstupu o polovinu řádu filtru
   g) Zpoždění lze kompenzovat **pouze** pomocí zero-pahase filtering (filtfilt)
   h) Řády filtrů se pohybují v desítkách až stovkách
   i) Řády filtrů se pohybují v jednotkách

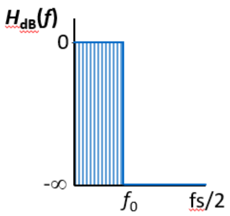
3) Nakreslete a matematicky zapište sinc() funkci. Nakreslete i její spektrum.

$$sinc(t) = \frac{\sin(t)}{t}; t \in \pm\infty$$

$$\lim_{t\to 0} \frac{\sin(t)}{t} = 1$$



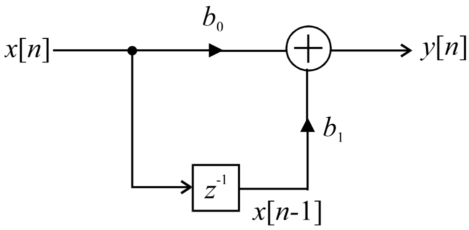
Ideální dolní propust



4) Pomocí sinc() funkce navrhněte digitální filtr typu horní propust (HP) řádu M=20, jehož mezní kmitočet f_0=20 Hz a vzorkovací kmitočet f_s=50 Hz. Zapište v MATLAB kódu.

```
M=20; f0=20; fs=50;
t=linspace(-(M/2)/fs,(M/2)/fs,M+1); % časová osa
sinc=sin(2*pi*f0*t)./(2*pi*f0*t); % sin(wt)/(wt)
sinc(M/2+1)=1; % lim(t=0)=1
A=fs/(2*f0); % korekce zesílení
sinc=sinc/A;
sinc=sinc.*hamming(length(sinc))'; % převážení oknem, DP

d=zeros(1,length(sinc));
d(M/2+1)=1; % jednotkový impulz

b=d-sinc; % impulz-BP=HP
```

5) Zakreslete do mřížkové struktury FIR diferenciátor, zapište jeho diferenční rovnici, pomocí z-transformace nalezněte pozici nul a pólů a odhadněte přenosovou charakteristiku.



$y[n] = x[n] - x[n\text{-}1]$

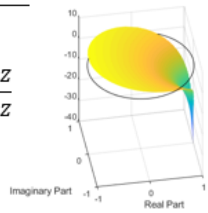
$Y(z) = b_0X(z) + b_1z^{-1}X(z)$

$b_0 = 1 \quad b_1 = -1$

$$H(z) = \frac{Y(z)}{X(z)} = \frac{b_0X(z) + b_1z^{-1}X(z)}{X(z)}$$

$$H(z) = \frac{b_0 + b_1z^{-1}}{1} = \frac{1 - z^{-1}}{1} \quad | \cdot \frac{z}{z}$$

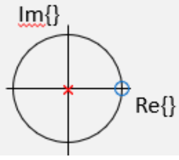
$$H(z) = \frac{z - 1}{z}$$

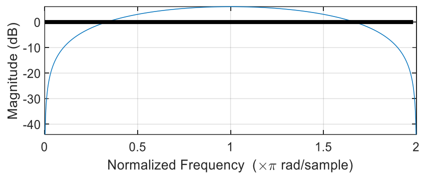


Kde leží póly a nuly:

Nuly $H(z)=0$ pro $z=1+0j$

Póly $H(z)=\infty$ pro $z=0+0j$





Otázky k opakování: IIR

1) Co znamená zkratka IIR? Jak souvisí řád filtru, koeficienty filtru a délka impulzní odezvy

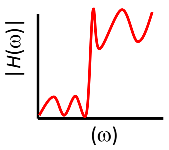
IIR – infinite impulse response (nekonečná impulzní odezva)

Po průchodu jednotkového impulzu LTI systémem se na jeho výstupu objeví impulzní odezva, která je nekonečná, protože systém obsahuje zpětné vazby vracející výstup opět na vstup. U stabilních systémů je imp. odezva tlumená. Řád IIR filtru definuje počet zpětnovazebních větví a_k.

2) Označte správné tvrzení o IIR filtrech
   a) Struktura filtru obsahuje pouze dopředné vazby
   b) Struktura může obsahovat pouze zpětné vazby
   c) Struktura může obsahovat dopředné i zpětné vazby
   d) Lineární zpoždění je v případě symetrické/antisymetrické impulzní odezvy
   e) Lineární zpoždění je v případě nesymetrické impulzní odezvy
   f) Zpoždění se kompenzuje posunem výstupu o polovinu řádu filtru
   g) Zpoždění lze kompenzovat **pouze** pomocí zero-pahase filtering (filtfilt)
   h) Řády filtrů se pohybují v desítkách až stovkách
   i) Řády filtrů se pohybují v jednotkách

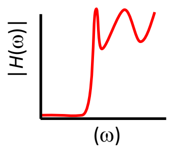
3) Zakreslete modul přenosové charakteristiky pro <u>horní propust</u> čtveřice nejpoužívanějších aproximací IIR filtrů. Popište základní vlastnosti filtrů (zvlnění, strmost, doba tlumení impulzní odezvy).

Eliptické (Cauerovi)



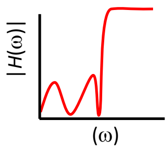
Nejstrmější
Nejdéle kmitá
Zvlněný v obou

Chebyshev Typ I



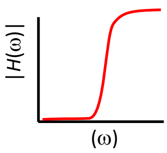
Strmý
Středně kmitá
Zvlněný v propustném

Chebyshev Typ II



Strmý
Středně kmitá
Zvlněný v zádržném

Butterworth



Maximálně plochý
Nejméně kmitá
Plochý

4) Zakreslete do mřížkové struktury IIR sumátor s integrační konstantou k, zapište jeho diferenční rovnici, pomocí z-transformace nalezněte pozici nul a pólů a odhadněte přenosovou charakteristiku.

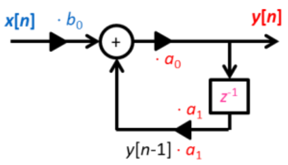
$$a_0 y[n] + a_1 y[n-1] = b_0 x[n]$$

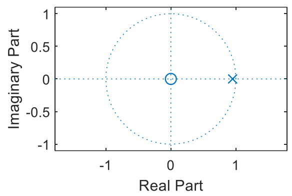
$$a_0 y[n] = b_0 x[n] - a_1 y[n-1]$$

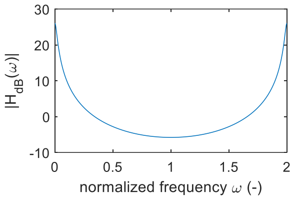
$$a_0 = 1; a_1 = -k; b_0 = 1$$

$$y[n] = x[n] + k y[n-1]$$

$$H[z] = \frac{Y[z]}{X[z]} = \frac{X[z] b_0}{(a_0 + a_1 z^{-1})} \cdot \frac{z}{z} = \frac{b_0 z}{a_0 z + a_1} = \frac{z}{z-k}$$







5) Jak se kompenzuje nelineární zpoždění filtrů. Blokově zakreslete postup. Jaké jsou výhody a nevýhody metody?

Zero-phase filtering, v MATLAB filtfilt

signál -> IIR filtrace -> otečení signálu v čase -> IIR filtrace -> otečení signálu v čase

V: ruší fázový posun, dvojnásobný útlum filtru (dvojnásobná filtrace)

N: impulzní odezva filtru před i za impulzem, výpočetní náročnost

ZZS08 – Převzorkování, obálka signálu

Úvod:

Převzorkování:

Digitální signály jsou vzorkovány různým vzorkovacím kmitočtem, takže vzájemné porovnání rozdílných časových řad není možné. Proto je nutno signály převzorkovat na stejný kmitočet. Nesmíme ovšem zapomínat, že při snížení vzorkovacího kmitočtu (decimaci) se posouvá i mezní kmitočet dle vzorkovacího teorému. Naopak zvýšením vzorkovacího kmitočtu (interpolaci) pouze prokládáme vzorky a nedochází ke změně nesené informace.

V praxi se setkáváme se signály, jejichž vzorkovací kmitočet je použit zbytečně vysoký. Např. EKG signál obsahuje užitečné složky v pásmu maximálně do 200 Hz, proto je plně dostačující vzorkovací kmitočet 500-1000 Hz, ačkoliv současné levné A/D převodníky zvládají až stovky kHz. U takto vzorkovaných signálů je zbytečně vysoký datový tok a tedy i náročnost na ukládání, propustnost přenosové soustavy a výpočetní výkon. Při návrhu filtrů se zlomovým kmitočtem u okrajů pásma (f_0 vs. fs) je obtížné ne-li nemožné navrhnout ostrý a stabilní filtr. Proto je vhodné signály decimovat.

Decimační faktor $D=fs/fsdec$ musí být celočíselný násobek frekvencí, protože algoritmus výpoču vybírá každý D-tý vzorek původního signálu. Aby nedošlo k aliasingu, musí být signál před decimací frekvenčně omezen anti-aliasing dolní propustí (AA-DP). Aby byl filtr dostatečně strmý, volíme přibližně $f_0=fsdec/(2.5$ až $3)$. Při decimačních poměrech $D>10$ není možné již navrhnout AA-DP, proto je třeba decimaci rozdělit do menších decimačních faktorů d: $D=d_1 \cdot d_2$.

Interpolace signálu na vyšší frekvenci je opět dán celočíselným faktorem $I=fsint/fs$. Algoritmus v prvním kroku vkládá mezi vzorky (I-1) nul. Budoucí interpolovaný signál tak obsahuje sekvenci impulzů o amplitudě původního signálu. Přefiltrováním impulzů dolní propustí (digital to analog converter DAC) s mezním kmitočtem $f_0=fs/2$ dojde na každém impulzu k impulzní odezvě – interpolaci. Ideálním typem DAC filtru je FIR: $\operatorname{sinc}(2\pi f_0 t) = \frac{\sin(2\pi f_0 t)}{2\pi f_0 t}$. Pokud interpolační faktor $I>10$, je nutno v zhledem k návrhu DAC rozložit iterpolaci do více kroků: $I=i_1 \cdot i_2$.

U neceločíselných poměrů převzorkování $R=fs/fsnew$ je potřeba nejprve signál interpolovat a následně decimovat. Jednotlivé faktory musejí být celočíselně dělitelné tak, aby R=I/D. U nepříznivém poměru kmitočtů s malým společným dělitelem, je třeba extrémně vysoce interpolovat a následně extrémě vysoce decimovat. Tento postup je vysoce paměťově a výpočetně náročný. Př.: Převzorkování z 1250 Hz na 512 Hz je nutno interpolovat faktorem $I=256$ (8×8×4) a decimovat faktorem $D=625$ (5×5×5×5).

Autoregresní modelování:

Při autoregresním modelování můžeme nastavit koeficienty a tak, aby impulzní odezva systému odpovídala modelovanému signálu $\hat{x}[n] = \sum_{i=0}^{M} x[n-i]a_i$. Tím získáváme IIR filtr, jehož přenosová funkce kopíruje spektrum signálu (odhad $\sqrt{PSD}$). Záměnou IIR koeficientů a za b ($b=a$) realizujeme FIR filtr, jehož charakteristika je komplementární a selektivně tlumí složky signálu.

Určení velikosti řádu modelu je vždy kompromisem mezi přesností popisu signálu a složitostí výpočtu. Vysoký řád věrně modeluje signál, nicméně za cenu výpočetní náročnosti a požadavku velkého množství vzorků pro predikci. Vysoký řád modelu také vede LPC model ke snaze predikovat i šumovou složku, což je nežádoucí. Pro odhad řádu modelu se využívají informační kriteriální podmínky. Odhad LPC modelu je optimalizační úloha, při kterém jsou hledány koeficienty IIR filtru a

tak, aby rozdílová složka signálu a predikovaného signálu byla co nejmenší $e[n]$=‖$x[n]$−$\hat{x}[n]$‖. S rostoucí složitostí modelu (řádem) klesá chyba predikce $E = \sum_n e[n]$. Kriteriální podmínky penalizují klesající chybu estimace *E* řádem modelu, která ji v ideálním případě mění na konvexní funkci. Minimum funkce pak odpovídá optimálnímu řádu modelu. Existuje mnoho kritérií, nejužívanějšími jsou Akaikeho informační kritérium (AIC) a Minimum description length (MDL): $AIC = N\ln(E) + 2M$; $MDL = N\ln(E) + M\ln(N)$; *N* je délka signálu, *M* je řád modelu.

Cíle:

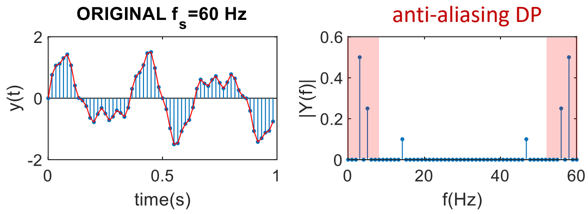
1) Filtrace izolinie EKG signálu:
   - <u>Extrahujte isolinii a odečtěte ji od EKG:</u>
   - Postupně decimujte signál pro spolehlivý návrh DP
   - Filtrací decimovaného signálu získejte isolinii
   - Interpolujte isolinii na původní *fs*
   - Odečtěte isolinii od EKG
2) Odhad spektra autoregresním modelováním:
   - Pomocí LPC zjistěte koeficienty autoregresního modelu
   - Zobrazte přenosovou funkci a určete frekvenci formantů u samohlásek
   - Odhadněte řád modelu pomocí informačního kritéria
   - **BONUS:** určete 1. a 2. formant u všech samohlásek a porovnejte s 1. zápočtovým úkolem

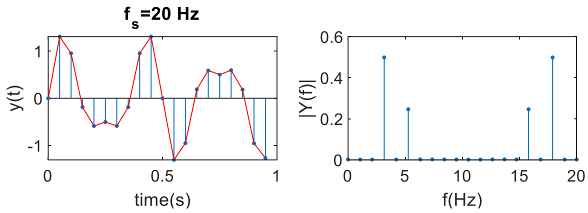
Užitečné funkce: `resample, filtfilt, lpc, freqz, roots, zplane, sort, audiowrite`

Nápověda:

Vytvořte si syntetický signál s třemi harmonickými složkami. Proveďte decimaci a interpolaci signálu. Vykreslete spektra signálů a uvědomte si roli anti-aliasing filtru a DAC filtru.

```
% syntetický signál
fs=60;
T=1;
t=linspace(0,T-1/fs,round(T*fs));
x=sin(2*pi*3*t)+0.5*sin(2*pi*5*t)+0.2*sin(2*pi*14*t);

... % zobrazte signál a spektrum
stem(t,x,'.');
hold on
plot(t,x,'r');
...
S=fft(x)/length(x);
F=linspace(0,fs,length(S));
stem(F,abs(S),'.');
...
```

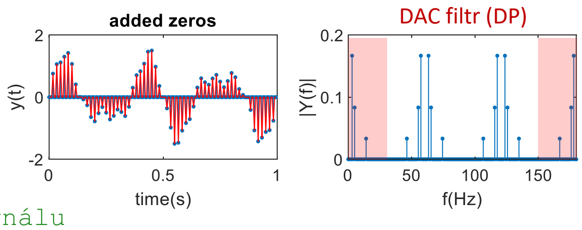




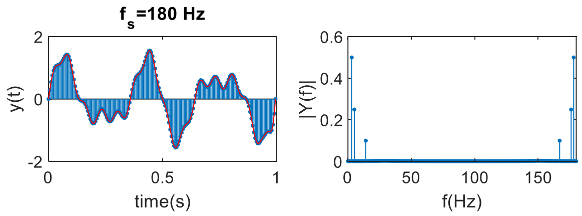

```
% decimujte signál na fsdec=20 Hz
fsdec=20; % nová fs
D=fs/fsdec; % decimační faktor

% anti-aliasing filtr
f0=round(fsdec/2.5); % f0<fsdec/2
[b,a]=butter(5,...); % IIR DP pro fs

xf=filtfilt(b,a,x); % omezení spektra
xd=xf(1:D:end); % každý D-tý vzorek
td=t(1:D:end); % časový vektor
... % zobrazte signál a spektrum
```

Kam zmizela složka 14 Hz?...

Co se stane s 14 Hz složkou, pokud nepoužijete DP?...



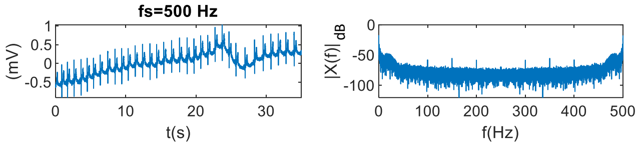

```
% interpolujte signál na fsint=180 Hz

fsint=180; % nová fs
I=fsint/fs; % interpolační faktor

xi=zeros(1,I*length(x)); % nová délka signálu
xi(1:I:end)=x; % impulzy z původního signálu (expandér)
ti=linspace(0,(length(xi)-1)/fsint,length(xi));
... % zobrazte signál a spektrum
```



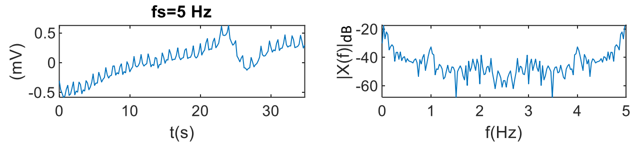

```
% navrhněte DAC filtr pomocí sinc()
% konvoluční maska = impulzní odezva
M=30;
f0=fs/2;
T=(M+1)/fsint; % sinc() pro novou fs
tsinc=linspace(-T/2,T/2,M+1);
% sinc()
bdac=sin(2*f0*pi.*tsinc)./(2*f0*pi.*tsinc);
bdac(ceil(end/2))=1;
bdac=bdac(:).*hamming(length(bdac));

% interpolovaný signál:
xii=filtfilt(bdac,...,xi); % koeficienty DAC a=? (FIR)
K=sqrt(sum((x.^2)/length(x))/sum((xii.^2)/length(xii)));
xii=K*xii; % korekce amplitudy (Parsevalova rovnost)
... % zobrazte signál a spektrum
```

Filtrujte izolinii jejím odečtením od původního EKG signálu.

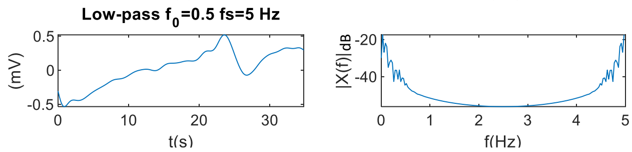
```
ekg=load('ECG_fs500Hz_iso.txt');
fs=500;
f0=0.5; % mezní kmitočet isolinie
...
% zobrazte signál a spektrum v dB:
plot(t,ekg);
...
```



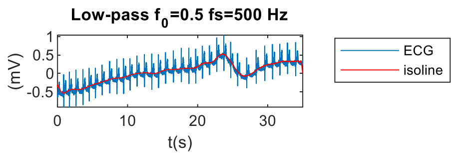

```
% decimace
fsdec=10*f0; % volíme cca 10*f0 (5 Hz)

R=fs/fsdec; % R=100 -> dvojstupňová 10x10
dec=resample(ekg,...,fs); % 500->50
dec=resample(dec,fsdec,...); % 50->5
tdec=linspace(0,(length(dec)-1)/fsdec,length(dec));
% zobrazte signál a spektrum v dB:
plot(tdec,dec);
...
```



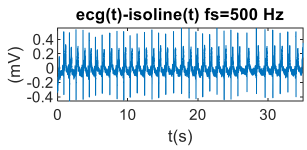

```
% navrhněte DP pro isolinii f0=0.5 Hz
[b,a]=butter(...);
iso=filtfilt(b,a,dec);
% zobrazte signál a spektrum v dB:
plot(tdec,iso);
...
```




```
% zpětná interpolace
iso_int=resample(iso,...,fsdec);
iso_int=resample(iso_int,fs,...);
...
plot(t,ekg);
hold on
plot(t,iso_int,'r');
```

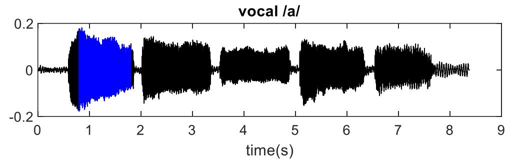


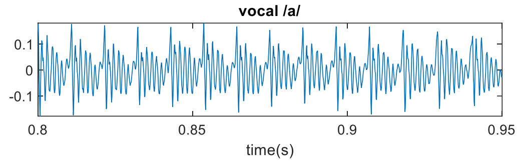

```
% odečtěte isolinii od signálu
ekg_f=ekg-iso_int;
... % vykreslete filtrovaný signál
```

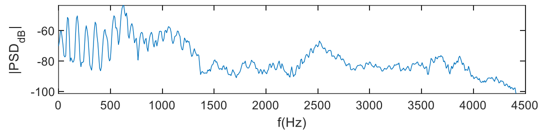


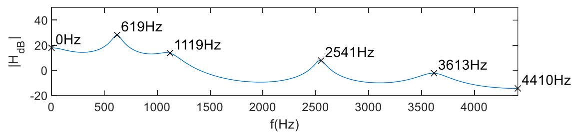
Odhad spektra autoregresním modelováním:

Hlasový trakt je buzen hlasivkami o základní frekvenci f_0 (hlasový tón). Rezonátory hlasového traktu (formanty) modulují hlasivkový tón a dochází k tvorbě znělých hlásek /a/, /e/, /i/, /o/, /u/. Pomocí LPC (IIR filtru) lze nalézt přenosovou charakteristiku, přičemž póly přenosové funkce korespondují s formantovými frekvencemi. Přenosová charakteristika modelovaného systému je aproximací spektra signálu, lze ji tedy využít pro odhad spektra (jako Welchova metoda pomocí FFT). Vyšší 3. a 4. formanty zasahují do frekvencí <4 kHz (u dětí <6 kHz). Proto signál decimujeme, čímž se zbavíme neužitečných složek.

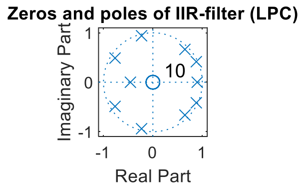









```
[vocal,fs]=audioread('AEIOU.m4a');
vocal=mean(vocal,2); % stereo->mono

vocal=resample(...); % decimujte
celočíselně signál na fs≈(2x4,5kHz), tj.
např. z 44100 Hz na 8820Hz. U vysokých
hlasů 2x6kHz.
fs=...; % nový vzorkovací kmitočet fs
t=...; % čas dle nového fs

% hláska /a/
va=vocal(round(0.8*fs):round(1.8*fs));
pwelch(...);
% vykreslete decimovaný signál, odhad PSD

M=10; % řád modelu nejprve odhadneme, ze
znalosti spektra 4 formanty + 1 sklon,
spektrum je symetrické tj. 2x(4+1);
a=lpc(va,M); % autoregresní model
[H,f]=freqz(1,a,[],fs,'whole'); % přenosová funkce
~ odhad spektra. Vykreslete v dB.
...
xlim([0 fs/2])
```



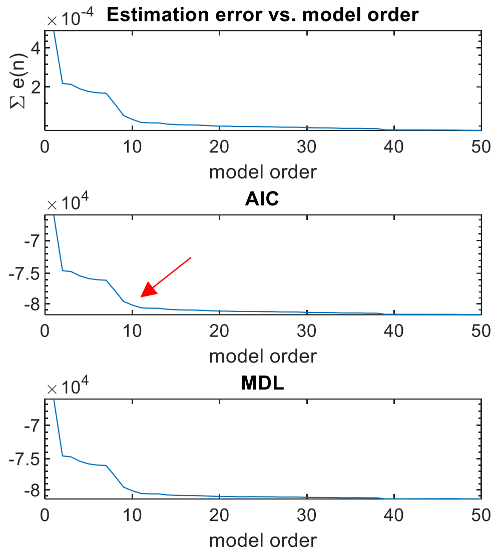

```
zplane(1,a); % zobrazte pozici pólů a nul,
uvědomte si, jak by vypadal membránový diagram a k němu přenosová funkce
IIR filtru

% najděte kořeny přenosové funkce H(ω), z normovaného kmitočtu ω vypočtěte
frekvence odpovídající pólům
rts=roots(a); % kořeny
wn=atan2(imag(rts),real(rts)); % normovaný kmitočet (úhel v rad)
fa=...; % norm. kmitočet na f (ω0=2πf0/fs)v radiánech
fa=sort(fa(fa>=0)); % pouze kladné frekvence (0-fs/2)
% Zobrazte zjištěné frekvence 1.-4. formantu do grafu a tabulky
hold on
for i=1:length(fa)
    idx=find(f>=fa(i),1); % indexy, kde fa odpovída frekvenční ose f
    plot(f(idx),20*log10(H(idx)),'kx') % značka křížku
    text(f(idx),20*log10(H(idx)),[' ' num2str(round(fa(i))) 'Hz'],...
        'VerticalAlignment','bottom') % textový popisek
end
```

formanty:	f_1 [Hz]	f_2 [Hz]	f_3 [Hz]	f_4 [Hz]
/a/				

Řád modelu nelze vždy předem určit, existují však techniky na jejich odhad, tzv. kriteriální podmínky. Postupně odhadujte LPC model a chybu estimace v závislosti na řádu modelu *M* a délce signálu *N*. Zvolte nižší optimální řád (*aic* nebo *mdl*).

```
Mmax=50; % maximální řád, např. 50
g=zeros(1,Mmax);
aic=zeros(1,Mmax);
mdl=zeros(1,Mmax);
for M=1:Mmax
     [~,E(M)]=lpc(va,M);
     aic(M)=...;  % AIC = Nln(E) + 2M
     mdl(M)=...;  % MDL = N ln(E) + Mln(N)
end
% vykreslete E(M), AIC(M), MDL(M) v log měřítku
plot(E); set(gca,'YScale','log');
...
% optimální řád modelu
[~,M_aic]=min(aic);
[~,M_mdl]=min(mdl);
```

Pozn.: V případě, že informační kritéria nejsou konvexní funkcí, zvolte ručně řád modelu v místě zlomu funkce (např. u AIC *M*=10)



BONUS:

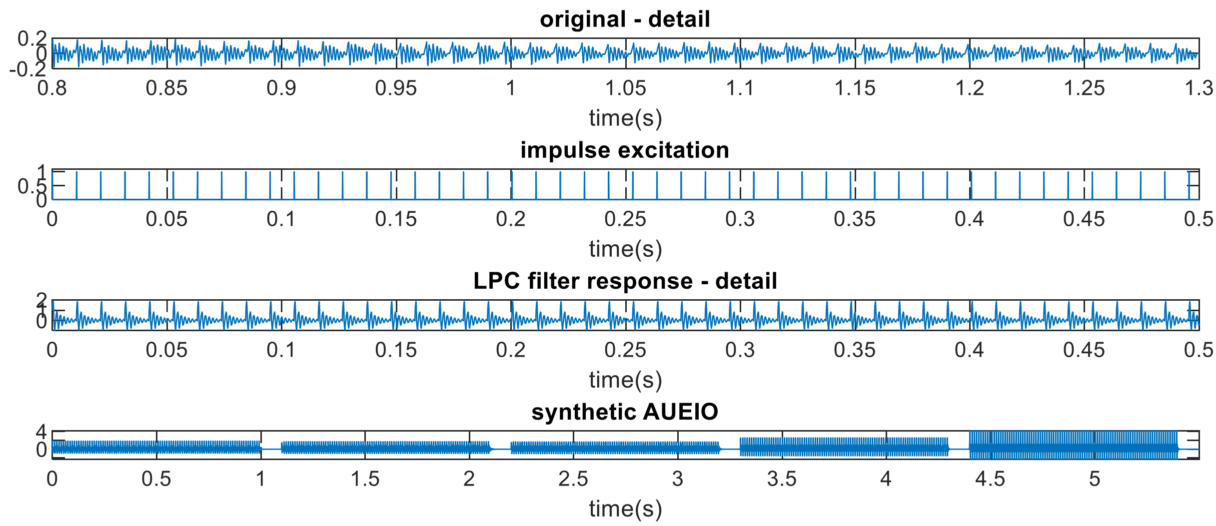
Pomocí LPC odhadněte 1. a 2. formant zbývajících samohlásek. Řád modelu použijte stejný jako pro samohlásku /a/. Porovnejte s formantovými frekvencemi odečtenými ze spektrogramu (1. semestrální úkol).

```
a=lpc(...); % autoregresní model pro /a/,/e/,/i/,/o/,/u/
...
```

Samohláska	f_1 (Hz)			f_2 (Hz)		
	~ norma	spektrogram	lpc	~ norma	spektrogram	lpc
/a/	500-1100			1100-1500		
/e/	400-700			1600-2100		
/i/	200-600			1800-2800		
/o/	400-700			900-1200		
/u/	200-500			500-1200		

Vygenerujte si syntetický hlas. Pro každou samohlásku odhadněte AR model (koeficienty filtru a). Do filtru (LTI) posílejte jednotkové impulzy, jež simulují kmitání hlasivek, tj. rozestup mezi impulzy odpovídá periodě základní hlasivkové frekvence $T_0=1/f_0$.

```
% pro jednoduchost si vyberte samohlásky stejné délky (1 s) a uložte do
matice aeiou (t x samohláska)
va=vocal(round(0.8*fs):round(1.8*fs));
... %ve,vi,vo
vu=vocal(round(6.6*fs):round(7.6*fs));
aeiou=[va,ve,vi,vo,vu];

M=10; % zvolte řád modelu, typicky 10
f0=95; % nastavte hlasivkovou frekvenci v Hz

% vygenerujte budící „hlasivkový" signál série jednotkových impulzů
s rozestupem T0=1/f0
buz=zeros(...,1); % délka buzení bude odpovídat délce (době) syntetizované
promluvy jednotlivé samohlásky
buz(...)=1; % na každou hlasivkovou periodu T0 vložte jednotkový impulz

synth=[];
for i=1:5 % pro každou samohlásku v aeiou
    a=lpc(aeiou(:,i),M); % koeficienty LPC (IIR filtr samohlásky)
    synth=[synth; filter(...)]; % vybuďte IIR filtr a přidejte za předchozí
    Pozn.: mezi synth. samohlásky můžete vložit i nulový vektor jako pauzu
end
sound(synth,fs); % záznam lze uložit audiowrite, vyžaduje převzorkování
```



Otázky k opakování: Parametrizace

1) Vyjmenujte alespoň tři parametry popisující rozkmit amplitudy.

Intenzita, energie, směrodatná odchylka, inter-kvartil rozsah, |min-max|

2) Vyjmenujte alespoň tři parametry popisující dominantní frekvenci signálu

Maximální spektrální čára, zero-crossing, mediánová frekvence, 1. spektrální moment, okamžitá frekvence Hilbertovi transformace

3) Napište parametry pro popis stochastického signálu a vysvětlete pojmy:

s normální distribucí:

střední hodnota: průměrná hodnota = stejnosměrná složka

směrodatná odchylka: průměr kvadrátu odchylek od stejnosměrné složky (energie signálu bez stejnosměrné složky)

s nenormální distribucí:

modus: nejčastěji se vyskytující hodnota

medián: hodnota, která rozděluje distribuce na dvě stejně velké (početné) části $p_{0.5}$

percentily: kvartily ($p_{0.25}$, $p_{0.75}$ / q_1, q_3), decily ($p_{0.1}$ ... $p_{0.9}$ / d_1, d_9), mezikvartilový rozsah ($p_{0.75}$ - $p_{0.25}$ / q_3-q_1)

strmost/šikmost: převaha nižších/vyšších hodnot nad vyššími/nižšími

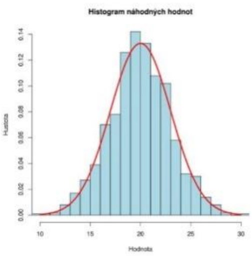
4) Jak spolu souvisí histogram, hustota pravděpodobnosti a kumulativní distribuční funkce?

Histogram znázorňuje distribuci měřené veličiny s konečným počtem prvků, které jsou rozděleny do stejně velkých intervalů. Počet prvků v intervalech je znázorňován v podobě sloupcového grafu.

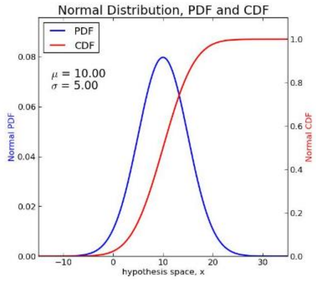
Histogram je diskrétní aproximací distribuční funkce (hustoty pravděpodobnosti), která spojitě vykresluje pravděpodobnost výskytu měřené veličiny.

Kumulativní distribuční funkce je integrálem hustoty pravděpodobnosti z leva a určuje, kolik procent z proměnných leží pod konkrétní hodnotou proměnné.

Histogram + hustota pravděpodobnosti



hustota p. + kumulativní d. funkce



5) V MATLAB pseudokódu napište postup pro výpočet počtu průchodů nulou

odečtení stejnosměrné složky	x=x-mena(x);
polarita + kladná nula	x=sign(x); x(x==0)=1;
počet sestupných a náběžných hran	n=sum(diff(x)~=0);

6) V MATLAB pseudokódu napište postup výpočtu mediánové frekvence

Jednostranné absolutní spektrum

```
N=length(x);
X=abs(fft(X))/N; F=linspace(0,N-fs/N,N)
X=X(1:end/2); F=F(1:end/2)
```

Kumulativní distribuční funkce

```
CDF=cumsum(X)/sum(X);
```

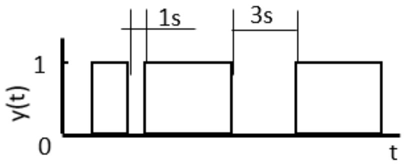
Medián

```
Fmed=F(find(CDF>=0.5,1));
```

7) Slovně popište algoritmus detekce R-špiček v EKG signálu

Filtrace 1-45 Hz
Zvýraznění rychlých změn diferenciací
Energie diferencovaného signálu
Výpočet energetické obálky dif. signálu
Prahování obálky, určení ROI s R-špičkou
Určení začátků a konců úseků ROI
Procházení ROI a nalezení lokálních maxim ve filtrovaném EKG signálu.

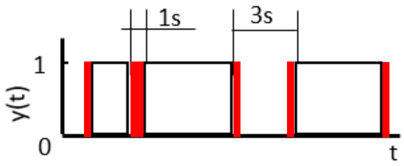
8) Co jsou morfologické operace? Navrhněte postup, jak vyplnit mezery pouze kratší než 1s v obdélníkovém signálu.

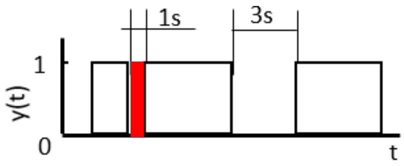


Dilatace: rozšíření do okolí

Eroze: smrsknutí

Postup zaplnění mezer: dilatace o 0.5 s, poté eroze o 0.5 s





Otázky k opakování: Jasové a geometrické transformace

1) Jak z barevného obrázku uděláme černobílý (šedoškálový).

Jasová hodnota černobílého obrazu je průměrem červené, zelené a modré vrstvy.
mean(image,3)

2) Vyjmenujte alespoň čtyři jasové transformace a uveďte postup výpočtu.

inverze (negativ): 1-image (normalizovaně 0-100%), 255-image (uint8)
prahování (black and white): image>práh, image<práh, k-means segmentace
úprava jasu: image +/- jas a následná ořez mimo interval 0-1 (0-255)
úprava kontrastu: a*image+b, a>1 (zvýšení), a<1 (snížení), b=0.5(1-a);
gama korekce: $image^{gamma}$
ekvalizace histogramu: převodní charakteristika == kumulativní distribuční funkci
redukce barev: rovnoměrná, k-means

3) Co a k čemu je ekvalizace histogramu? Napište postup v MATLAB pseudokódu.
Ekvalizace histogramu je jasová úprava k maximálnímu využití dynamického rozsahu obrázku ve prospěch řastěji se vyskytujících intenzit. Po ekvalizaci dochází k rovnoměrnějšímu využití jasových hodnot.

```
H=hist(image(:),256); % image v jasové škále 0-1
CDF=cumsum(H)/sum(H);
image_new=CDF(round(255*image+1));
```

4) Určete správné pořadí kroků v algoritmu k-menas.
   a) iterace pro aktualizaci těžišť
   b) zastavení při nalezení centroidů
   c) inicializace těžiště každé třídy
   d) definice počtu hledaných tříd
   e) přiřazení prvků k těžišti dle minimální vzdálenosti
   f) aktualizace těžiště jako průměr prvků ve třídě

   d, c, e, f, a, b

5) Vyjmenujte základní operace afinní transformace.

zrcadlení
změna měřítka
posun
zkosení
rotace

Otázky k opakování: Jasové a geometrické transformace

1) Jak z barevného obrázku uděláme černobílý (šedoškálový).

Jasová hodnota černobílého obrazu je průměrem červené, zelené a modré vrstvy.
mean(image,3)

2) Vyjmenujte alespoň čtyři jasové transformace a uveďte postup výpočtu.

inverze (negativ): 1-image (normalizovaně 0-100%), 255-image (uint8)
prahování (black and white): image>práh, image<práh, k-means segmentace
úprava jasu: image +/- jas a následná ořez mimo interval 0-1 (0-255)
úprava kontrastu: a*image+b, a>1 (zvýšení), a<1 (snížení), b=0.5(1-a);
gama korekce: $image^{gamma}$
ekvalizace histogramu: převodní charakteristika == kumulativní distribuční funkci
redukce barev: rovnoměrná, k-means

3) Co a k čemu je ekvalizace histogramu? Napište postup v MATLAB pseudokódu.

Ekvalizace histogramu je jasová úprava k maximálnímu využití dynamického rozsahu obrázku ve prospěch řastěji se vyskytujících intenzit. Po ekvalizaci dochází k rovnoměrnějšímu využití jasových hodnot.

```
H=hist(image(:),256); % image v jasové škále 0-1
CDF=cumsum(H)/sum(H);
image_new=CDF(round(255*image+1));
```

4) Určete správné pořadí kroků v algoritmu k-menas.
   a) iterace pro aktualizaci těžišť
   b) zastavení při nalezení centroidů
   c) inicializace těžiště každé třídy
   d) definice počtu hledaných tříd
   e) přiřazení prvků k těžišti dle minimální vzdálenosti
   f) aktualizace těžiště jako průměr prvků ve třídě

d, c, e, f, a, b

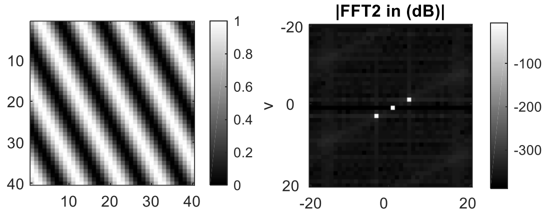
5) Vyjmenujte základní operace afinní transformace.

zrcadlení
změna měřítka
posun
zkosení
rotace

Otázky k opakování: Filtrace a korelace v obraze

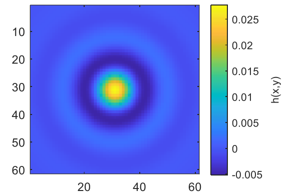
1) Co zobrazuje spektrum 2D Fourierovy transformace obrazu?

   Spektrum obrázku zobrazuje harmonické složky textury.



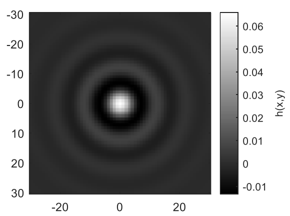
2) Jak vypadá impulzní odezva 2D filtru?

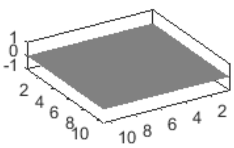
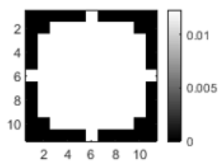
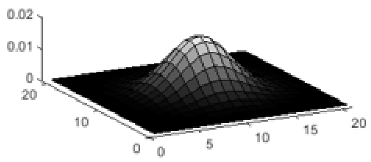
   Jedná se o 2D obraz (konvoluční masku)



3) Jakým způsobem se provádí filtrace 2D obrazu
   a) Konvolucí obrázku a masky
   b) Nulováním ve spektru

4) Uveďte příklady masek filtru (impulzních odezev) typu dolní propust (rozmazání)
   a) sinc (nevhodná)
   b) MA čtvercová
   c) MA kruhová
   d) Gaussovská



5) Uveďte příklady masek filtru (impulzních odezev) typu horní propust (zaostření hran)
   a) kruhová diference
   b) směrová diference

gradient v 45°

```
B = [-1 -1 -1;
     -1  9 -1;
     -1 -1 -1];
```

```
B = [-0.5   0;
      0    0.5]
```

6) Uveďte rozdíl mezi 2D konvolucí a 2D korelací
   Algoritmus výpočtu je stejný, pouze konvoluční maska je zrcadlená kolem středu oproti korelační masce.