



باسمه تعالی

دانشگاه صنعتی شریف

دانشکده مهندسی برق

۲۵۷۴۲ گروه ۳ - سیگنال و سیستم - نیمسال دوم  
۱۴۰۲-۰۳

## تمرین کامپیوتری سری یک

موعد تحویل: مطابق CW

## نحوه تحویل تمرین:

- گزارش تمرین خود را در قالب یک فایل *pdf*. تحویل دهید. در گزارش لازم است تمامی خروجی‌ها و نتایج نهایی، پرسش‌های متن تمرین و توضیح مختصری از فرآیند حل مسأله‌ی خود در هر قسمت را ذکر کنید.
  - کد کامل تمرین را در قالب یک فایل *m/.mlx/.py/.pyw/.ipynb*. تحویل دهید. لازم است بخش‌های مختلف تمرین در *section*های مختلف تفکیک شوند و کد تحویلی منظم و دارای کامنت‌گذاری مناسب باشد. بدیهی است آپلود کردن کدی که به درستی اجرا نشود، به منزله‌ی فاقد اعتبار بودن نتایج گزارش شده نیز می‌باشد.
  - توابعی را که (در صورت لزوم) نوشته‌اید، حتما در انتهای کد ضمیمه کنید و از ایجاد فایل جدای *m/.py*. خودداری کنید.
  - نام گذاری فایل‌های تحویلی را به صورت *HW01\_StudentNumber.pdf/m/.mlx/.py/.zip/.rar/...* انجام دهید.
- معیار نمره دهی:

- ساختار مرتب و حرفه‌ای گزارش
- استفاده از توابع و الگوریتم‌های مناسب
- پاسخ به سوالات تئوری و توضیح روش‌های مطلوب سوال
- کد و گزارش خروجی کد برای خواسته‌های مسأله

## نکات تکمیلی:

- در صورت داشتن هرگونه ابهام با آیدی تلگرام [@hasTSam](https://t.me/hasTSam) ارتباط برقرار کنید.
- اصولاً مشورت دانشجویان در حل تمرین‌ها نه تنها نکوهیده نیست، بلکه شدیداً توصیه می‌شود؛ اما توجه کنید که مشورت و رونویسی متفاوتند! لذا نتایج متفاوتی نیز دارند. همچنین برای مشورت در حل تمرین‌ها، از گروه درس استفاده کنید تا حق کسی در این روند ضایع نشود.
- شرافت انسانی ارزشی به مراتب والاتر از تعلقات دنیوی دارد. رونویسی تمارین، زیر پا گذاشتن شرافت خویشتن است؛ به کسانی که شرافتشان را زیر پا می‌گذارند هیچ نمره‌ای تعلق نمی‌گیرد.
- زمان تحویل تمرین تمدید نخواهد شد.

## ۱ کانولوشن

## ۱.۱ محاسبه ی کانولوشن

۱) با توجه به آنچه در کلاس آموختید، درباره ی روش محاسبه ی کانولوشن توضیح دهید و کانولوشن دو سیگنال زیر را محاسبه کنید و پاسخ را رسم کنید. همچنین استدلال کنید که کانولوشن دو آرایه یکی به طول  $n$  و دیگری به طول  $m$  چه طولی در نهایت خواهد داشت.

$$x_1[n] = \left(\frac{1}{3}\right)^n (u[n] - u[n-10])$$

$$x_2[n] = 0.9^n (u[n] - u[n-5])$$

۲) تابعی بنویسید که کانولوشن دو سیگنال دلخواه را محاسبه کند (مجاز به استفاده از توابع آماده ی متلب نیستید)

۳) یکی از روش های محاسبه کانولوشن دو آرایه ، استفاده از ضرب ماتریسی است. برای این کار آرایه اول را  $n$  در  $1$  و آرایه دوم را  $1$  در  $m$  در نظر میگیریم. همانطور که میدانید در اثر ضرب این دو ماتریس، ماتریس  $m \times n$  خواهیم داشت. ماتریس حاصل را در جهت قطر فرعی ماتریس جمع میکنیم. (مانند معادلات زیر) تابعی بنویسید که این عملیات را برای دو تابع دلخواه انجام دهد.

$$A * B = [a_1 a_2 \dots a_n] * [b_1 b_2 b_3 \dots b_m] = F$$

$$A^T B = \begin{bmatrix} a_{0,0} & \dots & \dots & a_{0,k} \\ \dots & \dots & a_{1,k-1} & \\ \dots & \dots & & \\ a_{k,0} & & & \\ & & & a_{n,m} \end{bmatrix}$$

$$F_k = \text{sum}(a_{0,k}, a_{1,k-1}, \dots, a_0)$$

for example :  $F_0 = a_{0,0}$

۴) حال با استفاده از توابع آماده ی متلب و تابعی که در بخش ۲ و ۳ برای محاسبه ی کانولوشن استفاده کرده اید کانولوشن دو سیگنال داده شده در بخش ۱ را محاسبه و باهم مقایسه کنید.

## ۲.۱ فیلترینگ و کرنل ها

(۱) در مورد کرنل های استفاده شده برای فیلتر کردن عکس های مختلف تحقیق کنید و سه مورد از کرنل هایی که راجبشان مطالعه کردید را در گزارشتان بیاورید.

تشخیص لبه (edge detection) یک تکنیک پردازش تصویر برای یافتن مرزهای اشیاء درون تصاویر است که با تشخیص ناپیوستگی در روشنایی کار می کند. تشخیص لبه برای تقسیم بندی تصویر و استخراج داده ها در زمینه هایی مانند پردازش تصویر، بینایی کامپیوتر و بینایی ماشین استفاده می شود.

(۲) از چه کرنلی برای edge detection استفاده میشود؟ توضیح دهید.

(۳) تابع  $f$  را در زیر در متلب رسم کنید و سپس فیلتر edge detection که کرنل آن برای شما مشخص شده است را بر روی تابع گفته شده اعمال کنید

$$f(x) = \begin{cases} 0, & |x| > 10 \\ -x, & 5 < |x| < 10 \\ x, & |x| < 5 \end{cases}$$

$$\text{one dimensional edge Kernel} = [1 \ 0 \ -1]$$

(۴) همانطور که از نتیجه ی بخش قبل متوجه شدید، کرنل داده شده برای پیدا کردن خطوط قائم به کار میرود. این کرنل را روی تصویر داده شده (zebra.jpg) اعمال کنید و مشاهدات خود را در گزارش بیاورید (دقت کنید باید از کرنل دو بعدی استفاده کنید).

(۵) حال میخواهیم کل خطوط تصویر داده شده را بیابیم برای این کار باید از کرنل هایی استفاده کنیم که تمام جهت هارا پوشش دهد. برای هر جهت یک کرنل داریم. برای جهت افقی، عمودی، قطری، قطر فرعی کرنل تعریف میشود. کرنل های زیر را در اختیار دارید، توضیح دهید هر کدام مربوط به پیدا کردن کدام یک از جهت ها میباشد.

$$A_1 = \begin{bmatrix} 1 & 0 & -1 \\ 1 & 0 & -1 \\ 1 & 0 & -1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & -1 & -1 \\ 1 & 0 & -1 \\ 1 & 1 & 0 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 0 & -1 \\ 0 & -1 & -1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} 1 & 1 & 1 \\ 0 & 0 & 0 \\ -1 & -1 & -1 \end{bmatrix}$$

۶) در این گام آغاز به edge detection میکنیم. برای اینکار مراحل زیر را دنبال کنید.

- ابتدا کرنل خطوط قائم را بر روی عکس اعمال کنید و در *convresult1* ذخیره کنید.
- حال کرنل خطوط افقی را بر روی عکس اعمال کنید و در *convresult2* ذخیره کنید.
- کرنل خطوط قطری را بر روی عکس اعمال کنید و در *convresult3* ذخیره کنید.
- کرنل خطوط قطر فرعی را بر روی عکس اعمال کرده و در *convresult4* ذخیره کنید.

با توجه به اینکه خطی که در راستای کرنلی است که کانولوشن آن روی عکس گرفته شده، قطعا مقدار بیشتری نسبت به کانولوشن کرنل های دیگر دارد عکس را باز سازی کرده و نشان دهید.

## ۲ تبدیل لاپلاس

## ۱.۲ آشنایی با تبدیل لاپلاس

۱. تبدیل لاپلاس توابع زیر را بدست آورده و سپس با استفاده از متلب از درستی پاسخ خود اطمینان حاصل کنید.

$$x_1(t) = (t \sin(t) + e^{-2t})u(t)$$

$$x_2(t) = t^2 e^{-5t} u(t)$$

$$x_3(t) = \cos(2t)u(t)$$

$$x_4(t) = t^2 \sinh(2t)u(t)$$

۲. وارون تبدیل لاپلاس توابع زیر را محاسبه کنید و با استفاده از متلب از درستی پاسخ خود اطمینان حاصل کنید.

$$F_1(s) = \frac{1}{s(s+2)(s+3)}$$

$$F_2(s) = \frac{10}{(s+1)^2(s+3)}$$

$$F_3(s) = \frac{2s^2}{(s^2+1)(s-1)^2}$$

## ۲.۲ تحلیل سیستم با استفاده از تبدیل لاپلاس

۱. پایداری سیستم های زیر را مشخص کنید.

$$H_1(s) = \frac{1}{s^5 + 125s^4 + 100s^3 + 100s^2 + 20s + 10}$$

$$H_2(s) = \frac{1}{s^6 + 5s^5 + 125s^4 + 100s^3 + 100s^2 + 20s + 10}$$

$$H_3(s) = \frac{5}{s(s^2 + 4s + 5)}$$

۲. پاسخ سیستم زیر را به ازای ورودی های زیر بیابید.

$$G(s) = \frac{1-s}{(s+1)(2s+1)}$$

ورودی ها:

$$x_1(t) = u(t)$$

$$x_2(t) = \sin(3t)u(t)$$

$$x_3(t) = e^{-0.5t}u(t)$$

۳.۲ حل معادله با استفاده از تبدیل لاپلاس

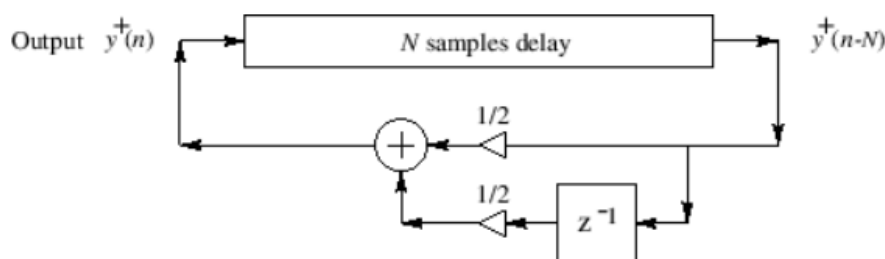
معادله ی زیر را در نظر بگیرید و با توجه به شرایط اولیه داده شده ( $y(0)=3$ ) پاسخ آن را بیابید.

$$\frac{y}{\partial t} + 2y = 12e^3$$

۳ تولید نوت های موسیقی با استفاده از یک سیستم گسسته

۱.۳ الگوریتم Karplus-Strong

Karplus-Strong الگوریتمی برای شبیه سازی صدای سیم (مثلا سیم های گیتار) است. این الگوریتم این کار را به کمک یک سیستم گسسته که بلوک دیاگرام آن را در شکل زیر مشاهده میکنید انجام می دهد. درباره ی این الگوریتم تحقیق کرده و توضیحی مختصر درباره ی آن در گزارش خود بیاورید.



۲.۳ تولید یک قطعه موسیقی

معادله تفاضلی توصیف کننده بلوک دیاگرام شکل بالا به صورت زیر می باشد (در بلوک دیاگرام ضریب آلفا برابر با یک در نظر گرفته شده است اما برای پایدار بودن سیستم، آن را مقداری مثبت و کمتر از یک در نظر میگیریم):

$$y[n] = x[n] + \alpha \frac{y[n-N] + y[n-(N+1)]}{2}$$

عملکرد الگوریتم به این صورت است که با قرار گرفتن سیگنال  $x[n]$  در ورودی سیستم که متشکل از  $N$  عدد رندوم می باشد (white noise)، در خروجی سیستم نت مورد نظر تولید خواهد شد. تنها نکته مهم انتخاب درست طول سیگنال ورودی است که همان دوره تناوب خواهد بود و می توان آن را به شکل زیر بدست آورد:

$$N = \text{floor}\left(\frac{Fs}{\text{freq}}\right)$$

که در این رابطه  $Fs$  برابر با فرکانس نمونه برداری می باشد. صداهایی که در دنیای واقعی شنیده می شوند سیگنال هایی پیوسته هستند، اما در دنیای دیجیتال امکان تولید صدای پیوسته وجود ندارد. به این منظور پارامتری تحت عنوان فرکانس نمونه برداری تعریف میکنیم که تعداد نمونه های سیگنال صوتی در یک ثانیه را نشان می دهد. برای تولید یک سیگنال صوتی گسسته، نمونه هایی از سیگنال پیوسته صوت بر میداریم که باتوجه له محدوده شنوایی و برای حفظ کیفیت نمونه برداری و جلوگیری از تداخل نمونه ها، این عمل را با  $Fs=44100 \text{ Hz}$  انجام می دهیم. با طی کردن مراحل زیر، تابع  $\text{generateNote}(\text{freq}, \text{duration}, \text{alpha})$  که در خروجی بردار نت مورد نظر با مدت دلخواه تولید میکند را بنویسید.

(۱) سیگنال  $x$  را به شکل برداری به طول  $N$  و اعضای بین ۱ و -۱ (با توزیع هم احتمال) تولید کنید.  
 (۲) با توجه به طول زمانی مورد نظر برای نت و باتوجه به فرکانس نمونه برداری، بردار  $y$  را با اعضای صفر به طول مناسب تولید کنید (این طول چقدر میشود؟)، سپس  $N$  مقدار اول آن را برابر با بردار  $x$  قرار دهید.  
 (۳) سیستم توصیف شده توسط معادله تفاضلی داده شده را پیاده سازی کنید و سیگنال  $y$  را به طور کامل و با طول مورد نظر ایجاد کنید.

پس از نوشتن این تابع، تابع  $\text{noteFreq}(\text{note})$  را بنویسید، به طوری که با گرفتن کاراکتر نت موسیقی، فرکانس آن را در خروجی بدهد. در [این لینک](#) می توانید فرکانس مربوط به هر نت (در اکتاو چهارم) را مشاهده کنید.

در پایان تابعی بنویسید که با گرفتن برداری از کاراکتر نت ها و برداری از مدت زمان هر نوت، بردار قطعه آهنگ مربوطه را تولید کند. خروجی آهنگ خود را برای بردارهای قطعه کد زیر بدست آورده و سپس به کمک دستور های `audioplayer` و `play` آن را پخش کنید. آهنگ ایجاد شده را به کمک دستور `au-diowrite` ذخیره کنید و در فایل تحویلی خود قرار دهید.

(قطعه کد ۱)

```

1 notes = ['G', 'G', 'A#', 'D#', 'D', ...
2         'G', 'G', 'A#', 'D', 'C', ...
3         'G', 'G', 'G', 'G', 'G', 'G#', ...
4         'G#', 'G#', 'G#', 'G#', 'G', 'G'];
5
6 noteDurations = [330, 330, 490, 490, 790, ...
7                  330, 330, 490, 490, 750, ...
8                  330, 330, 330, 490, 490, 700, ...
9                  330, 330, 330, 490, 490, 750];

```