



دانشگاه تهران

دانشکده علوم و فنون نوین

گزارش تمرین دوم

نام و نام خانوادگی	فاطمه چیت ساز
شماره دانشجویی	830402092
تاریخ ارسال گزارش	30 فروردین 1402

$$y(t) = 2x\left(\frac{t}{p}\right) - 1$$

دانشین ورودی $x_1(t) + x_2(t)$ به هم

$$y'(t) = 2\left(x_1\left(\frac{t}{p}\right) + x_2\left(\frac{t}{p}\right)\right) - 1$$

حالت اول: یک بار $x_1(t)$ را در $y(t)$ داریم، یک بار $x_2(t)$ را در $y(t)$ داریم

$$y''(t) = 2x_1\left(\frac{t}{p}\right) - 1 + 2x_2\left(\frac{t}{p}\right) - 1$$

حالت دوم: $y'(t)$ و $y''(t)$ را با هم جمع می‌کنیم و به دست می‌آوریم

$$y(t) = 2x\left(\frac{t}{p}\right) - 1$$

این زمان را t_0 می‌نامیم و داریم

$$y(t - t_0) = 2x\left(\frac{t - t_0}{p}\right) - 1$$

حالت سوم: $x(t - t_0)$ را با هم جمع می‌کنیم و داریم

$$y''(t) = 2x\left(\frac{t - t_0}{p}\right) - 1$$

حالت چهارم: $y''(t)$ را با $y'(t)$ جمع می‌کنیم و به دست می‌آوریم

$$y(t) = 2x\left(\frac{t}{p}\right) - 1$$

داشتن

درودی $x_1(t) + x_2(t)$ به هم

$$y'(t) = 2\left(x_1\left(\frac{t}{p}\right) + x_2\left(\frac{t}{p}\right)\right) - 1$$

حالت اولی با بار $x_1(t)$ درودی به هم دیار دینار $x_2(t)$ را درودی به هم

$$y''(t) = 2x_1\left(\frac{t}{p}\right) - 1 + 2x_2\left(\frac{t}{p}\right) - 1$$

مخرج $y'(t)$ و $y''(t)$ می باشد پس می باشد
بودن برقرار است با $\frac{t}{p}$ غیر خطی

$$y(t) = 2x\left(\frac{t}{p}\right) - 1$$

این زمان را t_0 به هم می باشد

$$y'(t - t_0) = 2x\left(\frac{t - t_0}{p}\right) - 1$$

حالت اولی درودی $x(t - t_0)$ با $\frac{t - t_0}{p}$ به هم می باشد

$$y''(t) = 2x\left(\frac{t - t_0}{p}\right) - 1$$

مخرج y' با y'' می باشد پس می باشد
بازمانده است

$$y[n] = n x[n]$$

چون به انداز دایره غرض باشد

دایره سیستم علی است

چون در هر لحظه به همان لحظه دایره است (به گذشته و آینده)

دایره غرض باشد در لحظه

$$y(t) = \int_{-\infty}^t x(\tau) d\tau$$

۱. با توجه به اینکه راهی در دسترس نیست رابطه کلی است
و با انداختن یک حاصل جمع داریم سیستم علی

$$y = \int_{-\infty}^t x(\tau - t_0) d\tau \quad \xrightarrow{\text{این میزان}} \int_{-\infty}^{t-t_0} x(\tau) d\tau$$

$$y'' = \int_{-\infty}^{t-t_0} x(\tau) d\tau$$

چون $y(t)$ و $y''(t)$ سیستم time invariant است

چون هر دو در هر زمان t تغییر می کنند در همان زمان و نسبت
دایره است علی

چون هر دو در لحظه t به یکدیگر نسبت دایره است
بجایگاه

$$y(t) = \sin(x(t)) \rightarrow$$

$$\left. \begin{array}{l} y_1(t) = \sin(x_1(t)) \\ y_2(t) = \sin(x_2(t)) \end{array} \right\} \rightarrow \sin(x_1(t) + x_2(t))$$

$$y'' = \sin(x_1(t) + x_2(t)) \quad \text{دادن صریح } x_1(t) + x_2(t)$$

چون $y' \neq y''$ رابطه خطی بودن برقرار نیست پس
سیستم غیر خطی

شیفت زمانی صریح $\sin(x(t-t_0)) \rightarrow y'$

شیفت زمانی خروجی $\sin(x(t-t_0)) \rightarrow y''$

چون $y' = y''$ پس سیستم خطی نامرئی

چون خروجی در هر زمان به مقدار همان لحظه وابسته

است پس سیستم ردین حافظه صلی است

$$y(t) = \begin{cases} x(t) + x(t-1) & t \geq 0 \\ 0 & 0 \leq t < 0 \end{cases}$$

چون رابطه بین ورودی ها خطی است و $x_1(t) + x_2(t)$ به سیستم

خروجی همان $y_1(t) + y_2(t)$ است خطی

ورودی در هر لحظه به همان لحظه وابسته

← علی و حافظه

انرژی مدی را سینتیم

$$y' \rightarrow \begin{cases} x(t-t_0) + x(t-t_0-2) & t \gg 0 \\ 0 & \text{و.و} \end{cases}$$

انرژی مدی را سینتیم

$$y'' \rightarrow \begin{cases} x(t-t_0) + x(t-t_0-2) & t-t_0 \gg 0 \\ 0 & \text{و.و} \end{cases}$$

چون $y' \neq y''$ ← time variant

$$y(n) = \frac{1}{2M+1} \sum_{k=-M}^{+M} x[n-k] \quad \text{ج.}$$

مدی در زمان n به $x[n+1]$ و ... است
است سینتیم ← سینتیم غیر مدی، دارای حافظه
است

اتحاد به اندیشه رابطه بین مدی و رابطه خطی است

مس رابطه خطی بودن برقرار است

$$y' = y_1 + y_2 = \frac{1}{2M+1} \sum x_1[n-k] + \frac{1}{2M+1} \sum x_2[n-k]$$

$$x_1 + x_2 \text{ مدی} \rightarrow y'' = \frac{1}{2M+1} \left(\sum (x_1[n-k] + x_2[n-k]) \right)$$

$$y' = y'' \rightarrow \text{خطی}$$

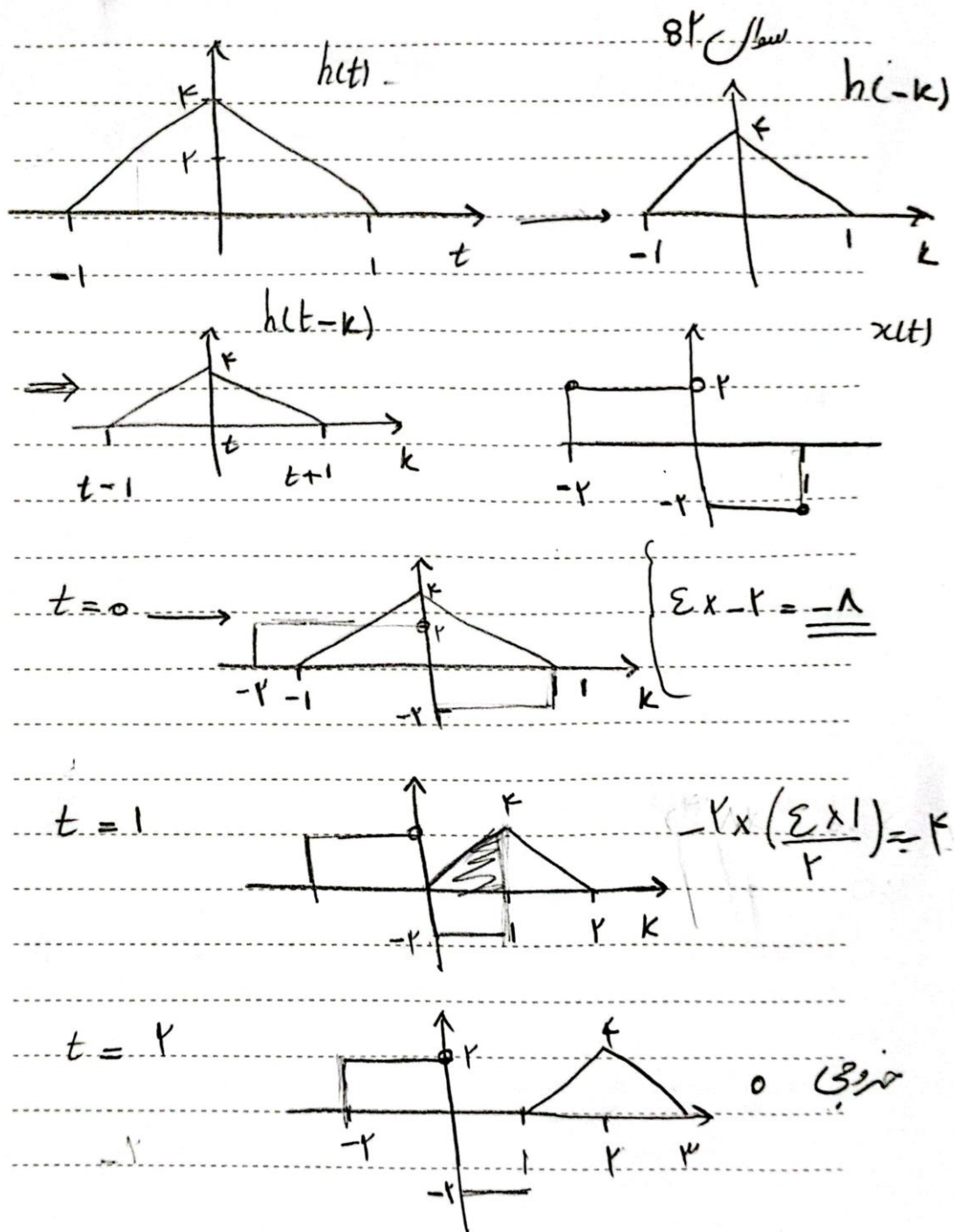
نشیب زمانی x $\rightarrow y'' = \frac{1}{2M+1} \sum x[n-k+t_0]$

نشیب زمانی y $\rightarrow y' = \frac{1}{2M+1} \sum x[n+t_0-k]$ **دانش**

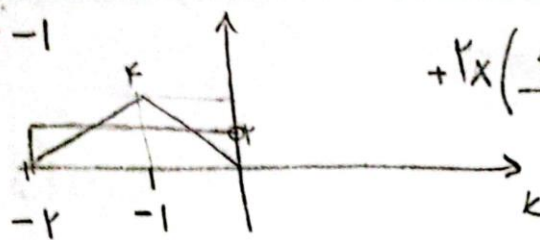
$y' = y'' \rightarrow$ تغییرناپذیری

Scanned with CamScanner

سوال دو



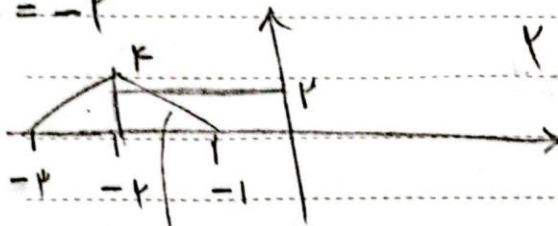
$t = -1$



$$1 \times \left(\frac{4 \times 2}{2} \right) = 1$$

دانش

$t = -2$



$$2 \times \frac{4 \times 1}{2} = 4$$

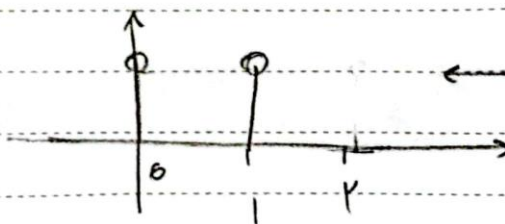
خروجی در بازه های $-2 < t \leq 1$ برابر با 4 است

خروجی در $t = -1$ برابر با 4 است

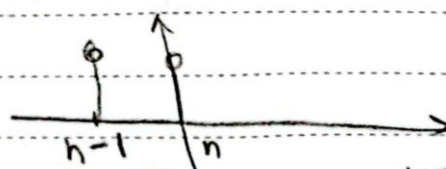
خروجی در $t = 1$ برابر با 4 است

مسئله 2 بخش بعدی:

$$h_r[n] * h_r[n] * h_r[n]$$



$$h_r[n-k]$$



$$\delta[n] + \delta[n-1] + \delta[n-2]$$

$$h_r[n] * h_r[n] = \begin{cases} n=0 \rightarrow 1 \\ n=1 \rightarrow 2 \\ n=2 \rightarrow 1 \end{cases}$$



$$h_1[n] * \text{دسته‌های} = y[n]$$

$$y[n] = \delta[n] + 0.5\delta[n-1] + 1.0\delta[n-2] + 1.1\delta[n-3] + 1.8\delta[n-4] + 1.4\delta[n-5] + \delta[n-6]$$

$$h_1[n] * (\delta[n] + 1.2\delta[n-1] + \delta[n-2]) = y[n]$$

فرض

$$H_1(\Omega) \times (1 + 1.2e^{-j\Omega} + e^{-2j\Omega}) =$$

$$Y(\Omega) \left((1 + 0.5e^{-j\Omega} + 1.0e^{-2j\Omega} + 1.1e^{-3j\Omega} + 1.8e^{-4j\Omega} + 1.4e^{-5j\Omega} + e^{-6j\Omega}) \right)$$

$$H_1(\Omega) = \frac{1 + 0.5e^{-j\Omega} + 1.0e^{-2j\Omega} + \dots}{1 + 1.2e^{-j\Omega} + e^{-2j\Omega}}$$

✓
حال با عکس
فرض

$$\frac{1}{(1 + e^{-j\Omega})^2} \frac{0.5e^{-j\Omega} + 1.0e^{-2j\Omega}}{1 + 1.2e^{-j\Omega} + e^{-2j\Omega}}$$

what about: ←

$$h_1[n] * (\delta[n] + 1.2\delta[n-1] + \delta[n-2]) = y[n]$$

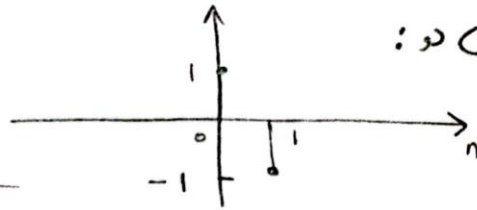
$$h_1[n] + 1.2h_1[n-1] + h_1[n-2] = y[n]$$

$$n=0 \quad h_1[0] + 1.2h_1[-1] + h_1[-2] = 1$$

$$n=1 \quad h_1[1] + 1.2h_1[0] + h_1[-1] = 0.5$$

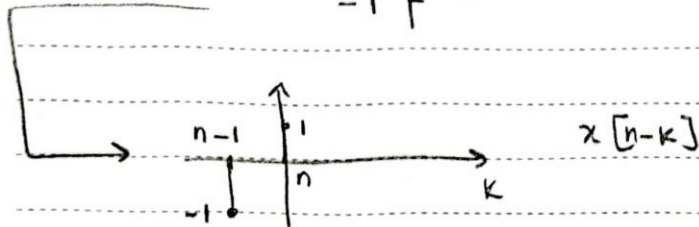
→ ??

$$x[n] =$$



خوبن لقمه خوبن دو:

دانش



نقشه

$$\begin{aligned} n=0 & \longrightarrow |x| = 1 \\ n=1 & \longrightarrow \omega x + -|x| = \underline{\underline{f}} \\ n=2 & \longrightarrow 10 - \omega = 0 \\ n=3 & \longrightarrow 11 - 10 = 1 \\ n=4 & \longrightarrow 1 - 11 = -10 \\ n=5 & \longrightarrow \Sigma - 1 = -4 \\ n=6 & \longrightarrow 1 - \Sigma = -10 \\ n=7 & \longrightarrow 0 - 1 = -1 \end{aligned}$$

نقشه

$$x[n] = 3x \omega^{-|n-2|}$$

سوال 8.3

$$n \geq 2 \longrightarrow x[n] = 3x \omega^{-n+2}$$

$$n < 2 \longrightarrow x[n] = 3x \omega^{n-2}$$

سوال سه:

$$x[n] = r^n \omega^{-|n-r|} \quad \text{سوال 8.3}$$

$$n \geq r \longrightarrow x[n] = r^n \omega^{-n+r}$$

$$n < r \longrightarrow x[n] = r^n \omega^{n-r}$$

$$x_1[n] = r_1 a^{(-n+r)} \quad \text{for } n \geq r$$

$$X_1(\omega) = \sum_{n=r}^{+\infty} r_1 a^{-n+r} e^{-j\omega n}$$



$$= r_1 a^r \sum_{n=r}^{\infty} \left(\frac{1}{a}\right)^n e^{-j\omega n}$$

$$\begin{array}{l} m = n+r \longrightarrow \\ \boxed{m+r=n} \end{array} \quad = r_1 a^r \times \sum_{m=0}^{\infty} \left(\frac{1}{a}\right)^r \times \left(\frac{1}{a}\right)^m e^{-j\omega m}$$

$$r_1 a^r \times \left(\frac{1}{a}\right)^r \times e^{-j\omega r} \sum_{m=0}^{\infty} \left(\frac{1}{a}\right)^m e^{-j\omega m}$$

$$m=0 \longrightarrow \text{Sigma is } 1$$

$$m=1 \longrightarrow \text{Sigma is } \frac{1}{a} \times e^{-j\omega}$$

$$m=r \longrightarrow \text{Sigma is } \left(\frac{1}{a}\right)^r \times e^{-j\omega r}$$

So Because of geometric Series

$$\sum_{m=0}^{\infty} \left(\frac{1}{a}\right)^m e^{-j\omega m} \longrightarrow \frac{a}{1-r}$$

a is First term = 1

$$r = \text{Common ratio} = \frac{1}{a} e^{-j\omega}$$

So $\frac{a}{1-r}$ is

Wednesday

چهارشنبه

2018

21 March

۱۴۳۹

۲ رجب

فروردین ۱۳۹۷

$$\frac{1}{1 - (\frac{1}{\omega}) \times e^{-j\omega}}$$

$$X_1(\Omega) = \sum_{n=-\infty}^{\infty} \omega^n \times \frac{1}{\omega} \times e^{-j\omega n} \times \frac{1}{1 - \frac{1}{\omega} e^{-j\omega}}$$

$$X_r(\Omega) = \sum_{n=-\infty}^{\infty} \omega^n \times e^{-j\omega n}$$

$$X_r(\Omega) = \omega \times \sum_{n=-\infty}^{\infty} e^{-j\omega n} \times \omega^n$$

$$x[n] = a^n \cos(\omega n + \phi) u[n]$$

$$X(\Omega) = \sum_{n=-\infty}^{\infty} x[n] e^{-j\omega n}$$

$$x[n] = a^n \times \frac{1}{2} (e^{j\omega n + \phi} + e^{-j\omega n + \phi}) u[n]$$

$u[n]$ is zero for $n < 0$ so

$$X(\Omega) = \sum_{n=0}^{\infty} \frac{1}{2} a^n (e^{j\omega n + \phi} + e^{-j\omega n + \phi}) e^{-j\omega n}$$

آغاز نوروز (تعطیل) - شهادت حضرت امام علی الرقی الهادی علیه السلام (۲۵۴ هـ.ق)

۲۰ ۲۹ ۲۸ ۲۷ ۲۶ ۲۵ ۲۴ ۲۳ ۲۲ ۲۱ ۲۰ ۱۹ ۱۸ ۱۷ ۱۶ ۱۵ ۱۴ ۱۳ ۱۲ ۱۱ ۱۰ ۹ ۸ ۷ ۶ ۵ ۴ ۳ ۲ ۱ هفته ۱

۲

پنجشنبه

Thursday

22 March

2018

فروردین ۱۳۹۷

۴ رجب

۱۴۳۹

خوبی توهم بازش بسف

$$\Rightarrow \frac{1}{2} \sum_{n=0}^{\infty} a^n e^{j(\omega_0 n + \phi)} x e^{-j\omega n} + \frac{1}{2} \sum_{n=0}^{\infty} a^n e^{-j(\omega_0 n + \phi)} x e^{-j\omega n}$$

$$\rightarrow \left\{ \frac{1}{2} e^{j\phi} * \sum_{n=0}^{\infty} (a e^{j(\omega_0 - \omega)})^n \right\} = \frac{1}{2} e^{j\phi} \frac{1}{1 - a e^{j(\omega_0 - \omega)}} \quad \text{مشتق اول}$$

$$a x e^{-j(\omega_0 - \omega)} \quad \text{مشتق دوم} \quad \text{مشتق اول فقط اینجا}$$

$$e^{j\phi} x \frac{1}{2} x \frac{1}{1 - a e^{-j(\omega_0 + \omega)}} \quad \text{مشتق دوم}$$

مشتق اول نهایی ما

عید نوروز (تعطیل) - هجوم مأموران ستم شاهی پهلوی به مدرسه فیضیه قم (۱۳۲۲ ه.ش) - آغاز عملیات فتح المبین (۱۳۶۱ ه.ش)

۳

جمعه

Friday

23 March

2018

۵ رجب

۱۴۳۹

مشتق دوم

$$X(\omega) = \sum_{n=-\infty}^{+\infty} v x e^{-j\omega n}$$

$$x[n] = 1 \quad \text{البتة فوري}$$

$$2 \leq \omega \leq 4 \quad \delta(\omega)$$

س. آبرین

$$14 \leq \omega \leq 21 \quad \delta(\omega)$$

عید نوروز (تعطیل)

Week

20	29	28	27	26	25	24	23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---	---	---	---	---	---	---	---	---

$$x[n] = A \cos(\omega_0 n + \phi)$$

$$X(\Omega) = \sum_{-\infty}^{\infty} x[n] e^{-j\Omega n}$$

Saturday

شنبه

2018

24 March

۱۴۳۹

۶ رجب

۱۳۹۷ فروردین

۸ d
۴

$$X(\Omega) = \sum_{-\infty}^{+\infty} A \left(e^{j(\omega_0 n + \phi)} + e^{-j(\omega_0 n + \phi)} \right) \times \frac{1}{r} x e^{-j\Omega n}$$

$$X(\Omega) = \frac{A}{r} \sum e^{j(\omega_0 n + \phi - \Omega n)} + \frac{A}{r} \sum e^{-j(\omega_0 n + \phi + \Omega n)}$$

$$\omega = \omega_0 \quad \text{or} \quad \omega = -\omega_0$$

$$\text{for } \omega = \omega_0 \rightarrow X(e^{j\omega_0}) = \frac{A}{r} e^{j\phi}$$

$$\text{for } \omega = -\omega_0 \rightarrow X(e^{-j\omega_0}) = \frac{A}{r} e^{-j\phi}$$

$$X(\Omega) = \frac{A}{r} x e^{j\phi} \delta(\omega - \omega_0) + \frac{A}{r} x e^{-j\phi} \delta(\omega + \omega_0)$$

$$x[n] = A \sin(\omega_0 n + \phi) (u[n] - u[n - 1])$$

تفاوت بین دو نمونه ۱ مقادیر غیر صفر

$$X(\Omega) = \sum_{-\infty}^{\infty} A \sin(\omega_0 n + \phi) e^{-j\Omega n}$$

$$X(\Omega) = \sum_{-\infty}^{\infty} A \left(e^{j(\omega_0 n + \phi)} - e^{-j(\omega_0 n + \phi)} \right) \times \frac{1}{j} e^{-j\Omega n}$$

عید نوروز (تعطیل)

۲۰ ۲۹ ۲۸ ۲۷ ۲۶ ۲۵ ۲۴ ۲۳ ۲۲ ۲۱ ۲۰ ۱۹ ۱۸ ۱۷ ۱۶ ۱۵ ۱۴ ۱۳ ۱۲ ۱۱ ۱۰ ۹ ۸ ۷ ۶ ۵ ۴ ۳ ۲ ۱

بخش دوم :

تابع `compute_dtft`: این تابع تبدیل فوریه زمان دیسکرت (DTFT) یک سیگنال را محاسبه می‌کند. کد از یک آرایه ورودی x و دو بردار n و w استفاده می‌کند. x نشان دهنده سیگنال ورودی است، n برداری از نقاط زمانی است که متناظر با مقادیر سیگنال x است و w برداری از فرکانس‌هایی است که می‌خواهیم برای DTFT محاسبه کنیم.

این کد در هر مقدار فرکانسی w با استفاده از یک حلقه به تمام نقاط زمانی n مربوطه می‌شود. برای هر مقدار w ، مقدار $e^{-j\omega n}$ برای تمامی نقاط زمانی محاسبه می‌شود و سپس جمع می‌شود تا مقدار نهایی DTFT به دست آید.

```
import numpy as np
def compute_dtft(x, n, w):
    # Computes Discrete-time Fourier Transform
    # [X] = compute_dtft(x,n,w)
    #
    # X = DTFT values computed at frequencies w
    # x is a finite-duration sequence over n
    # n is the vector of "time" values over which the computation is performed
    # w is a vector of frequencies used in the output
    X = np.zeros(len(w), dtype=complex)
    for k in range(len(w)):
        X[k] = np.sum(x * np.exp(-1j * w[k] * n))
    return X
```

خب در مرحله بعد توابعمون رو تعریف میکنیم (همون توابع صورت سوال)

```
# (a) x[n] = 3 * (5) ^ (- |n - 2|) for all n
def x_a(n):
    return 3 * (5 ** (-np.abs(n - 2)))

# (b) x[n] = alpha ^ n * cos(omega_0*n + phi) * u[n], |alpha| < 1
def x_b(n, alpha=0.9, omega_0=1, phi=0):
    return (alpha ** n) * np.cos(omega_0 * n + phi) * (n >= 0)

# (c) x[n] = 7 for all n
def x_c(n):
    return 7

# (d) x[n] = A * cos(omega_0*n + phi) for all n
def x_d(n, A=1, omega_0=1, phi=0):
    return A * np.cos(omega_0 * n + phi)

# (e) x[n] = A * sin(omega_0*n + phi) * (u[n] - u[n - 9])
def x_e(n, A=1, omega_0=1, phi=0):
    return A * np.sin(omega_0 * n + phi) * ((n >= 0) & (n < 9))
```

در مرحله بعد باید بیایم یک تابع بنویسیم که وقتی dtft را یافتیم نمودار را بر اساس مقدار و بر اساس مقادیر مختلف فاز نمایش دهد

```
def plot_dtft(w, X, title):

    plt.figure(figsize=(12, 6))
    plt.suptitle(title,)
    plt.subplot(1, 2, 1)
    plt.stem(w / (2 * np.pi), 20 * np.log10(np.abs(X)), use_line_collection=True)
    plt.title('Magnitude (dB)')
    plt.xlabel('Normalized Frequency (cycles/sample)')
    plt.ylabel('|X(e^{j\omega})| (dB)')

    plt.subplot(1, 2, 2)
    plt.stem(w / (2 * np.pi), np.angle(X), use_line_collection=True)
    plt.title('Phase')
    plt.xlabel('Normalized Frequency (cycles/sample)')
    plt.ylabel('\angle X(e^{j\omega})')
    plt.show()
```

حال زمان آن رسیده تا از توابعمون استفاده کنیم :

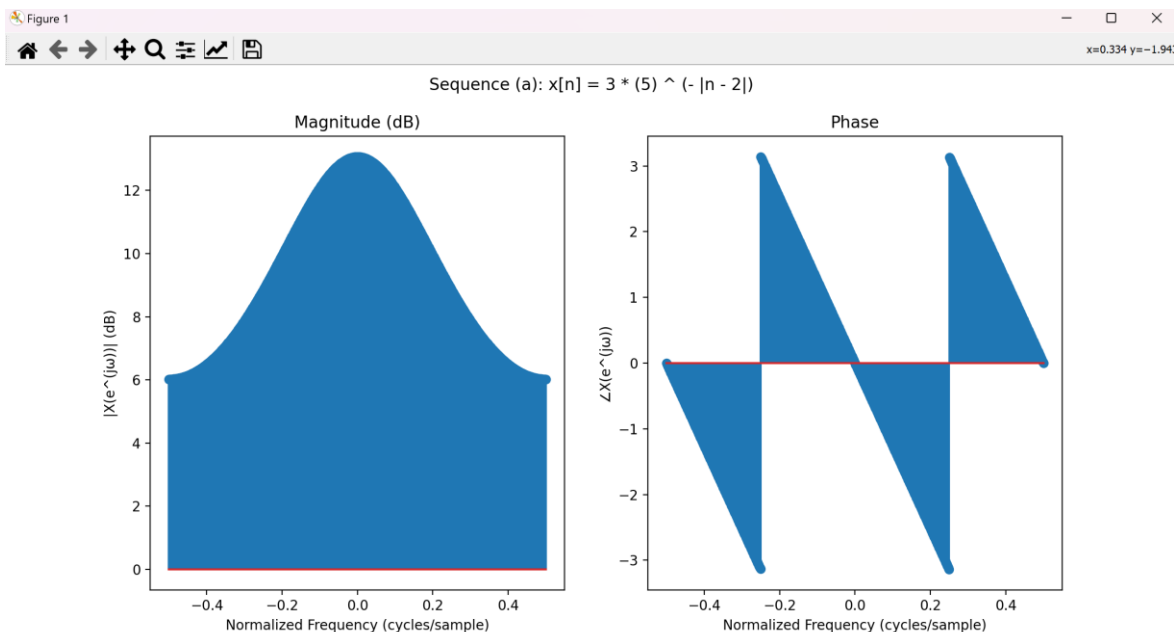
برای نمونه x_a : ما تعداد نمونه ها را بیست نمونه میگذاریم ک ده نمونه قبل صفر و ده نمونه بعد صفره و برای w هم بازه اش را تعیین میکنیم و میریم تو کار نمودار کشیدن

```
# Example usage for sequence (a)
N = 20
n = np.arange(-N, N, dtype=float)
w = np.linspace(-np.pi, np.pi, 1000)

x = x_a(n)
X = compute_dtft(x, n, w)
plot_dtft(w, X, "Sequence (a):  $x[n] = 3 \cdot (5)^{-|n-2|}$ ")

# Example usage for sequence (b)
```

خروجی برای x_a



در مرحله بعدی از ما خواسته شده است تا مرحله قبلی را با کتابخانه انجام دهیم برای این کار ما از کد زیر استفاده کرده ایم :

```
from scipy.signal import freqz
def plot_dtft_and_freqz(w, X, title, n, x):
    plt.figure(figsize=(12, 6))
    plt.suptitle(title)

    plt.subplot(1, 2, 1)
    plt.stem(w / (2 * np.pi), 20 * np.log10(np.abs(X)), use_line_collection=True)
    plt.title('Magnitude (dB)')
    plt.xlabel('Normalized Frequency (cycles/sample)')
    plt.ylabel('|X(e^(jw))| (dB)')

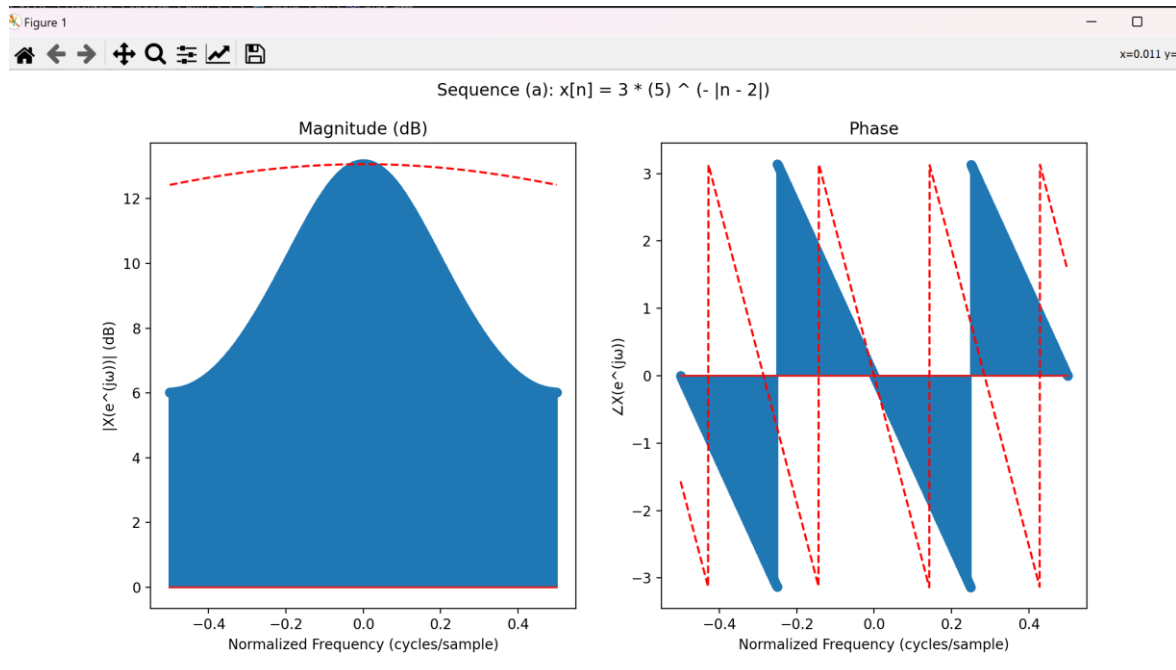
    plt.subplot(1, 2, 2)
    plt.stem(w / (2 * np.pi), np.angle(X), use_line_collection=True)
    plt.title('Phase')
    plt.xlabel('Normalized Frequency (cycles/sample)')
    plt.ylabel('∠X(e^(jw))')

    # Compute and plot freqz
    w_freqz, H_freqz = freqz(x, worN=w / (2 * np.pi))
    plt.subplot(1, 2, 1)
    plt.plot(w / (2 * np.pi), 20 * np.log10(np.abs(H_freqz)), 'r--')
    plt.subplot(1, 2, 2)
    plt.plot(w / (2 * np.pi), np.angle(H_freqz), 'r--')

    plt.show()
```

در اینجا کتابخانه خود را اضافه کرده و مقدار x را به آن میدهیم و دو نمودار را باهم رسم میکنیم تا بتوانیم همزمان خروجی ها را ببینیم :

نمونه خروجی برای xa:



چرا نتایج متفاوت ؟

اختلاف در خروجی‌ها ممکن است به دلایل مختلفی باشد. این دلایل ممکن است شامل موارد زیر باشد:

فرمول‌های محاسبه: ممکن است فرمول‌های استفاده شده در دو روش محاسبه متفاوت باشند. به عنوان مثال، ممکن است یکی از روش‌ها از فرمول محاسبه تبدیل فوریه دیسکرت (DTFT) مستقیم استفاده کند، در حالی که روش دیگر از الگوریتم‌های مختلفی مانند FFT برای محاسبه استفاده کند که ممکن است به دقت‌های مختلفی منجر شود.

پارامترهای ورودی: در برخی موارد، اختلافات ممکن است به دلیل مقادیر متفاوت پارامترهای ورودی باشد. مثلاً مقادیر آلفا، امپلیتود، فرکانس و فاز ممکن است در دو روش محاسبه متفاوت باشند.

یک دلیل دیگر اینه ممکنه نرمال سازی ها متفاوت باشه .

سوال چهار:

سوال پنج

در این گزارش، یک برنامه ساده برای ایجاد یک چت بات مشابه زیگپ را توضیح می‌دهیم. این برنامه از تکنولوژی‌های مختلفی مانند ضبط و تبدیل صدا به متن و هوش مصنوعی برای ایجاد تعامل با کاربر

استفاده می‌کند. در اینجا از کتابخانه‌های PyAudio برای ضبط صدا، هزار برای تبدیل صدا به متن فارسی، و از API gpt4free برای ایجاد پاسخ‌های چت بات استفاده شده است. برنامه علاوه بر اینکه به کاربر امکان می‌دهد تا از طریق گفتگو با صدا با چت بات تعامل داشته باشد، همچنین تمامی گفتگوها را در یک فایل متنی ذخیره می‌کند

ابتدا کتابخانه‌ها را نصب میکنیم

```
pip install PyAudio
```

Requirement already satisfied: PyAudio in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages>

```
pip install keyboard
```

Requirement already satisfied: keyboard in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages>
Note: you may need to restart the kernel to use updated packages.

```
pip install hezar[all]
```

```
pip install -U g4f[all]
```

Requirement already satisfied: g4f[all] in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (0.2.9.5)
Requirement already satisfied: requests in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: aiohttp in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: brotli in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: pycryptodome in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: curl-cffi>=0.6.2 in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: certifi in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])
Requirement already satisfied: browser-cookie3 in <c:\users\asus\anaconda3\envs\myenv\lib\site-packages> (from g4f[all])

کتابخانه pyaudio برای ضبط صدا کتابخانه کیبورد برای بررسی اینکه یوزر با کیبورد چه وارد کرده تا مثلا با توجه به آن ضبط صدا را قطع کنیم و کتابخانه hezar برای تبدیل صدای فارسی به متن فارسی و کتابخانه gpt4 به عنوان یک llm برای پاسخ به متن

1. تابع `record_audio` به عنوان ورودی شماره‌ی یک عدد صحیح `i` می‌گیرد. این تابع با استفاده از کتابخانه‌های `wave`، `pyaudio` و `keyboard`، صدا را ضبط می‌کند و در یک فایل wav ذخیره می‌کند. و تا زمانی که کلید q فشار داده نشود ضبط را ادامه می‌دهد

1. `i``: این پارامتر یک شماره صحیح است که به عنوان پسوند فایل ضبط شده استفاده می‌شود. ماین پارامتر برای جدا سازی فایل‌های ضبط شده از هم، از این روش استفاده می‌شود تا فایل‌ها با نام‌های مناسبی ذخیره شوند.

2. `p``: این متغیر یک نمونه از کلاس `pyaudio.PyAudio` است که برای مدیریت عملیات مربوط به پخش و ضبط صدا استفاده می‌شود.

3. `stream``: این متغیر یک شیء از کلاس `Stream`` است که برای باز و بسته کردن جریان صوتی استفاده می‌شود. تنظیمات جریان صوتی از طریق پارامترهای `CHANNELS``، `FORMAT`` و `RATE`` انجام می‌شود.

پارامتر های مهم «

4. `FORMAT``: این پارامتر نوع فرمت صوتی است که در ضبط و پخش صوت استفاده می‌شود.

5. `CHANNELS``: تعداد کانال‌های صوتی که برای ضبط استفاده می‌شود (مثلاً استریو یا مونو).

6. `RATE``: نمونه‌برداری (sampling) بر ثانیه که مشخص می‌کند چقدر اطلاعات از صدا در هر ثانیه ضبط می‌شود.

7. `CHUNK``: اندازه‌ی بلوکی از داده‌های صوتی که به صورت پشته‌ای از جریان صوتی خوانده می‌شود.

در این فانشن با عبارت `keyboard.is_pressed('q')` چک می‌کند که آیا کاربر دکمه "q" را فشرده است یا خیر، در صورتی که دکمه‌ی "q" فشرده شده باشد، حلقه‌ی ضبط متوقف می‌شود.

```
def record_audio(i):
    p = pyaudio.PyAudio()

    stream = p.open(format=FORMAT, channels=CHANNELS,
                    rate=RATE, input=True,
                    frames_per_buffer=CHUNK)

    print("در حال ضبط صدا. برای پایان دادن به ضبط صدا دکمه کیو را بفشارید")

    frames = []
    while True:
        data = stream.read(CHUNK)
        frames.append(data)
        if keyboard.is_pressed('q'):
            break

    print("ضبط صدا پایان یافت")

    stream.stop_stream()
    stream.close()
    p.terminate()
    wf = wave.open(f"{WAVE_OUTPUT_FILENAME}{i}.wav", 'wb')
    wf.setnchannels(CHANNELS)
    wf.setsampwidth(p.get_sample_size(FORMAT))
    wf.setframerate(RATE)
    wf.writeframes(b''.join(frames))
    wf.close()
```

2. تابع `audio_to_text` یک تابع پردازش گفتار به متن است که یک فایل صوتی را به عنوان ورودی می‌گیرد و متن معادل آن را استخراج می‌کند. این تابع دو گام اصلی دارد:

1. بارگیری مدل: ابتدا، تابع مدل مورد نیاز برای تبدیل گفتار به متن را بارگیری می‌کند. در اینجا، از مدلی با نام `"hezarai/whisper-small-fa"` استفاده شده است که یک مدل مربوط به تبدیل گفتار فارسی به متن است.

2. پیش‌بینی متن: سپس، تابع فایل صوتی را به عنوان ورودی به مدل می‌دهد تا متن معادل آن را پیش‌بینی کند. این متن معمولاً به صورت یک لیست از فریم‌ها یا قطعه‌های کوتاه صوتی همراه با متن معادل هر فریم یا قطعه برگردانده می‌شود.

3. استخراج متن: در نهایت، متن استخراج شده از فایل صوتی به عنوان خروجی تابع برگشت داده می‌شود. احتمالاً در این تابع، متن معادل با اولین فریم یا قطعه صوتی در نتایج برگشتی مدل، به عنوان متن نهایی انتخاب شده است.

```
def audio_to_text(audio_file):
    ... model = Model.load("hezarai/whisper-small-fa")
    ... transcripts = model.predict(audio_file)
    ... return transcripts[0]['text']
```

3. تابع `get_chatbot_response` : از یک کلاینت برای ارتباط با سرورهای G4F استفاده می‌کند و از مدل زبانی GPT-3.5 Turbo برای تولید پاسخ برای متن ورودی استفاده می‌کند. پس از ارسال متن به عنوان پیام کاربر به سرور، پاسخ تولید شده توسط مدل برای همان متن ورودی استخراج شده و به عنوان خروجی تابع ارسال می‌شود.

```
from g4f.client import Client

def get_chatbot_response(text):

    client = Client()
    response = client.chat.completions.create(
        model="gpt-3.5-turbo",
        messages=[{"role": "user", "content": text}],
    )

    return response.choices[0].message.content
```

4. تابع `run_chatbot` یک چتبات ساده را پیاده‌سازی می‌کند که به کمک گفتگوهای صوتی، پاسخ‌های چتبات را ایجاد می‌کند و مکالمه را چاپ می‌کند. عملکرد این تابع به شرح زیر است:

داخل یک حلقه بینهایت، این تابع ابتدا صدا را ضبط می‌کند، سپس آن را به متن تبدیل می‌کند و سپس با استفاده از تابع `get_chatbot_response` پاسخ مربوط به متن ورودی را از چتبات دریافت می‌کند.

سپس مکالمه (پیام کاربر و پاسخ چتبات) را به همراه نشان دادن "شما" و "بات" در هر پیام، چاپ می‌کند.

مکالمه را در یک فایل ذخیره می‌کند.

در پایان هر چرخه، از کاربر می‌پرسد که آیا می‌خواهد ادامه دهد یا خیر. اگر پاسخ مثبت باشد، ادامه می‌دهد و در غیر این صورت از حلقه خارج می‌شود.

```
def run_chatbot():
    i = 0
    while True:
        # Record audio
        record_audio(i)

        # Convert audio to text
        text = audio_to_text(f"{WAVE_OUTPUT_FILENAME}{i}.wav")

        # Get the chatbot's response
        response = get_chatbot_response(text)

        # Print the conversation
        conversation = f"شما: {text}\nبات: {response}\n---\n"
        print(conversation)

        # Save the conversation
        save_conversation(conversation)

        i=i+1
        # Ask user to continue or exit
        choice = input("[Y/N] آیا میخواهید ادامه دهید؟ ")
        print("\n---\n")
        if choice.upper() == 'N':
            break
```

ما این کار را برای چندین حالت مختلف از پارامترها انجام دادیم :

state 1

```
CHUNK = 1024
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 16000
```

```
# Audio recording settings
CHUNK = 1024
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 16000
WAVE_OUTPUT_FILENAME = "output"
run_chatbot()
```

حالت دوم :

state 2

CHUNK = 1024

FORMAT = pyaudio.paInt16

CHANNELS = 1

RATE = 8000

```
# Audio recording settings
CHUNK = 1024
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 8000
run_chatbot()
```

حالت سوم :

state 3

CHUNK = 512

FORMAT = pyaudio.paInt16

CHANNELS = 1

RATE = 16000

```
# Audio recording settings
CHUNK = 512
FORMAT = pyaudio.paInt16
CHANNELS = 1
RATE = 16000
run_chatbot()
```

حالت چهارم :

state 4

CHUNK = 1024

FORMAT = pyaudio.paInt16

CHANNELS = 2

RATE = 16000

```
# Audio recording settings
CHUNK = 1024
FORMAT = pyaudio.paInt16
CHANNELS = 2
RATE = 16000
run_chatbot()
```

پارامتر ها و تاثیر انها :

:CHUNK

CHUNK یا اندازه تکه‌های داده‌ای صوتی که از جریان ورودی خوانده می‌شوند است.

مقدار بزرگتر CHUNK باعث می‌شود بیشترین حجم داده‌ها به صورت همزمان خوانده شود، که می‌تواند باعث افزایش کارایی و کاهش نویز در ضبط صوتی شود. اما این باعث افزایش تأخیر در پردازش می‌شود. مقدار کوچک CHUNK باعث کاهش تأخیر در پردازش می‌شود، اما ممکن است باعث افزایش نویز در صدا شود و همچنین منجر به بارگذاری بیشتر CPU شود.

:FORMAT

FORMAT نوع فرمت داده‌ای است که از جریان ورودی صوتی استفاده می‌شود.

فرمت pyaudio.paInt16 یکی از فرمت‌های رایج است که از 16 بیت برای نمونه‌برداری صوتی استفاده می‌کند و اطلاعات صوتی را به صورت اعداد صحیح ذخیره می‌کند.

:CHANNELS

CHANNELS تعداد کانال‌های صوتی است که در ضبط صدا استفاده می‌شود (مثلاً استریو یا مونو).

استفاده از دو کانال برای CHANNELS به معنای ضبط صوت استریو است که اطلاعات صوتی را از دو منبع مختلف ضبط می‌کند. این می‌تواند تجربهٔ گوش دادن فعال‌تری ارائه دهد.

:RATE

RATE یا نرخ نمونه‌برداری، تعداد نمونه‌هایی است که در یک ثانیه از جریان ورودی صوتی خوانده می‌شود.

مقدار RATE بالا منجر به کیفیت بالاتر و در نتیجه فایل‌های صوتی بزرگتر می‌شود. اما نرخ نمونه‌برداری بالاتر می‌تواند باعث بارگذاری بیشتر CPU و حافظه شود.

اما در مثال ما خروجی تمام ست‌آپ‌ها یکی شده است که در پوشه‌های 1 state تا چهار ذخیره شده است

نمونه خروجی بعد سه پرسش و پاسخ :

```
در حال ضبط صدا. برای پایان دادن به ضبط صدا دکمه کیو را فشار دهید
ضبط صدا پایان یافت
شما: سلام
بات: سلام! چطور می‌توانم به شما کمک کنم؟
---

---

در حال ضبط صدا. برای پایان دادن به ضبط صدا دکمه کیو را فشار دهید
ضبط صدا پایان یافت
شما: حالت چطوره؟
بات: ممنونم که پرسیدید! حال شما چطور است؟ چطور می‌توانم به شما کمک کنم؟
---

---

در حال ضبط صدا. برای پایان دادن به ضبط صدا دکمه کیو را فشار دهید
ضبط صدا پایان یافت
شما: بهترین زبان برنامه‌نویسی چیه؟
بات: انتخاب بهترین زبان برنامه‌نویسی به ویژگی‌ها و نیازهای پروژه‌ها و تیم‌ها بستگی دارد. هر زبان برای کاربردهای خاصی مناسب‌تره. به عنوان مثال:

1. **Python**: برای توسعه وب، علوم داده، هوش مصنوعی، اپلیکیشن موبایل و غیره.
2. **JavaScript**: برای توسعه وب، به ویژه برنامه‌های تعاملی و رابط کاربری.
---
```

خروجی فایل txt:

```
File Edit View
شما: سلام
بات: سلام! چطور می‌توانم به شما کمک کنم؟
---
شما: حالت چطوره؟
بات: ممنونم که پرسیدید! حال شما چطور است؟ چطور می‌توانم به شما کمک کنم؟
---
شما: بهترین زبان برنامه‌نویسی چیه؟
بات: انتخاب بهترین زبان برنامه‌نویسی به ویژگی‌ها و نیازهای پروژه‌ها و تیم‌ها بستگی دارد. هر زبان برای کاربردهای خاصی مناسب‌تره. به عنوان مثال:

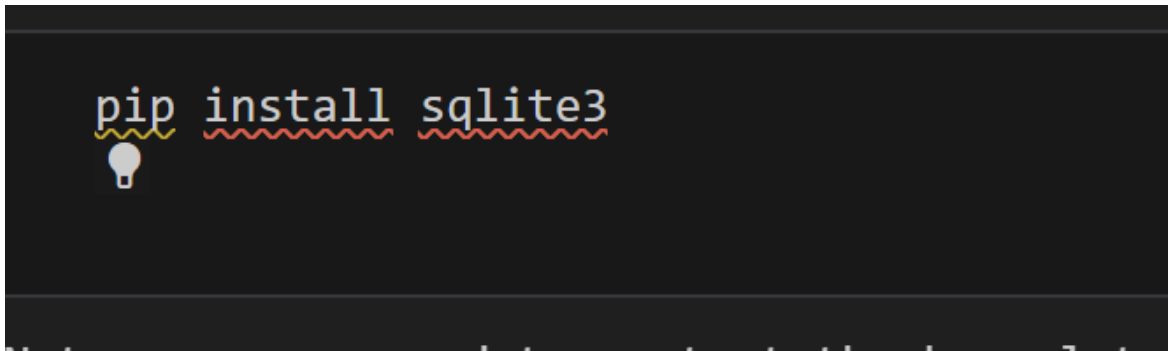
1. **Python**: برای توسعه وب، علوم داده، هوش مصنوعی، اپلیکیشن موبایل و غیره.
2. **JavaScript**: برای توسعه وب، به ویژه برنامه‌های تعاملی و رابط کاربری.
3. **Java**: برای توسعه برنامه‌های مستقل، وب و اندروید.
4. **C++**: برای توسعه برنامه‌های با کارایی بالا مانند بازی‌ها و نرم‌افزارهای سیستمی.
5. **Swift**: برای توسعه برنامه‌های iOS و macOS.
6. **Kotlin**: برای توسعه برنامه‌های اندروید.
7. **C#**: برای توسعه برنامه‌های ویندوز و بازی‌ها با استفاده از Unity.

همچنین، برای انتخاب بهترین زبان، می‌توانید به عواملی مانند اندازه پروژه، تیم توسعه، توانایی‌های فنی، و نیازهای کاری توجه کنید.
---
```

بخش دوم :

خروجی این قسمت در state5 است.

این برنامه از یک پایگاه داده SQLite برای ذخیره اطلاعات کاربران و گفتگوها استفاده می‌کند. برای اینکه برنامه به درستی کار کند، ابتدا باید SQLite را نصب کنیم



```
pip install sqlite3
```

- `register_user(username, password)` : این تابع یک کاربر جدید را در پایگاه داده ثبت می‌کند.

```
def register_user(username, password):  
    c.execute("INSERT INTO users (username, password) VALUES (?, ?)", (username, password))  
    conn.commit()
```

- `login_user(username, password)` : این تابع یک کاربر را با نام کاربری و رمز عبور ورودی احراز هویت می‌کند و شناسه کاربر را برمی‌گرداند.

```
def login_user(username, password):  
    c.execute("SELECT id FROM users WHERE username=? AND password=?", (username, password))  
    user_id = c.fetchone()  
    if user_id:  
        return user_id[0]  
    else:  
        return None
```

- `get_previous_conversations(user_id)` : این تابع گفتگوهای قبلی یک کاربر را از پایگاه داده بازیابی می‌کند.

```
def get_previous_conversations(user_id):  
    c.execute("SELECT request, response FROM conversations WHERE user_id=?", (user_id,))  
    return c.fetchall()
```

- `save_conversation(user_id, request, response)` : این تابع یک گفتگوی جدید را برای یک کاربر در پایگاه داده ذخیره می‌کند.

- `save_report(user_id, conversations)` : این تابع یک گزارش از گفتگوهای یک کاربر را در یک فایل متنی ذخیره می‌کند.

```
def save_report(user_id, conversations):  
    filename = f"{user_id}_conversations.txt"  
    with open(filename, "w", encoding="utf-8") as f:  
        for request, response in conversations:  
            f.write(f"User: {request}\n")  
            f.write(f"Bot: {response}\n\n")
```

2. تابع `main` :

- این تابع برنامه را اجرا می‌کند.

- ابتدا نام کاربری و رمز عبور را از کاربر دریافت می‌کند و با استفاده از توابع بخش دوم، احراز هویت کاربر را انجام می‌دهد.

- اگر کاربر معتبر باشد، گفتگوهای قبلی او را از پایگاه داده بازیابی کرده و نمایش می‌دهد.

- سپس در یک حلقه، صدا را ضبط کرده، به متن تبدیل می‌کند، پاسخ چت‌بات را دریافت می‌کند، گفتگو را در پایگاه داده ذخیره می‌کند و گزارش کلیه گفتگوهای کاربر را در یک فایل متنی ذخیره می‌کند.

- در پایان، تابع `save_report` فراخوانی می‌شود تا گزارش نهایی گفتگوهای کاربر در یک فایل ذخیره شود.

```
def main():
    username = input("نام کاربری: ")
    password = input("رمز عبور: ")
    user_id = login_user(username, password)
    if user_id:
        print("ورود با موفقیت انجام شد.")
        previous_conversations = get_previous_conversations(user_id)
        if previous_conversations:
            print("گفتگوهای قبلی:")
            for request, response in previous_conversations:
                print(f"شما: {request}")
                print(f"بات: {response}\n")

        while True:
            # Record audio
            i = 0
            record_audio(i)

            # Convert audio to text
            text = audio_to_text(f"{WAVE_OUTPUT_FILENAME}{i}.wav")

            # Get the chatbot's response
            response = get_chatbot_response(text)

            save_conversation(user_id, text, response)
            print(f"شما: {text}")
            print(f"بات: {response}\n")

            i=i+1
            # Ask user to continue or exit
            choice = input("[Y/N]؟ ادامه دهید؟ ")
            print("\n---\n")
            if choice.upper() == 'N':
                break

        save_report(user_id, previous_conversations + [(text, response)])
```

ذخیره چند نمونه یوزر :

```
register_user("fatemeh", "12345")
register_user("roya", "6789")
```

این برنامه اطمینان حاصل می‌کند که اطلاعات کاربر و گفتگوها به طور امن و دقیق در پایگاه داده ذخیره شده و همچنین گزارش مفصلی از فعالیت‌های کاربر در یک فایل متنی ذخیره می‌شود.

نمونه خروجی :

• ورود با موفقیت انجام شد.

گفتگوهای قبلی:

شما: سلام

بات: سلام! چطور می‌توانم به شما کمک کنم؟

در حال ضبط صدا. برای پایان دادن به ضبط صدا دکمه کیو را بفشارید

ضبط صدا پایان یافت

شما: حالت چگونه؟

بات: من خوبم، ممنون که پرسیدید! شما چطورید؟