RESPUESTAS DEL SISTEMA

NO. DE PRACTICA: 6		Fecha de entrega:	31 de octubre de 2023
INTEGRANTES DE EQUIPO: 19 NAVIL PINEDA RUGERIO	Practica:	Individual	En equipo 2 personas

OBJETIVO:

MATERIAL Y EQUIPO: **COMPUTADORA CON MATLAB**

DESARROLLO:

Paso 1. Calcula la Transformada Discreta de Fourier (DFT) de cada una de las siguientes secuencias x[n]

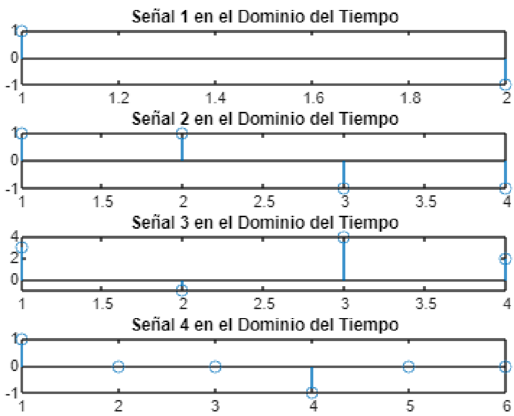
A) $\boldsymbol{x[n] = (1, -1)}$
B) $\boldsymbol{x[n] = (1, 1, -1, -1)}$
C) $\boldsymbol{x[n] = (3, -1, 4, 2)}$
D) $\boldsymbol{x[n] = (1, 0, 0, -1, 0, 0)}$

Para calcular la Transformada Discreta de Fourier, se utiliza la función de Matlab "fft", que recibe como parámetros una secuencia x[n], por ello cada una de las secuencias anteriores se escriben como vectores:

```
x1 = [1,-1];
x2 = [1,1,-1,-1];
x3 = [3,-1,4,2];
x4 = [1,0,0,-1,0,0];
```

Y se grafican las señales.

```
SUBPLOT(4,1,1)
stem(x1)
title("Señal " + num2str(1) + " en el dominio del tiempo")

subplot(4,1,2)
stem(x2)
title("Señal " + num2str(2) + " en el dominio del tiempo")

subplot(4,1,3)
stem(x3)
title("Señal " + num2str(3) + " en el dominio del tiempo")

subplot(4,1,4)
stem(x4)
title("Señal " + num2str(4) + " en el dominio del tiempo")
```



Después, se calcula su transformada con la función de Matlab, y por cada transformada se calcula su magnitud, aplicando el valor absoluto al resultado de la transformada:

```
% Señal 1-------------------------------------------------------------
y1 = fft(x1);
m1 = abs(y1); % magnitud
% Señal 2-------------------------------------------------------------
Y2 = FFT(X2);
m2 = abs(y2); % magnitud
% SEÑAL 3-------------------------------------------------------------
y3 = fft(x3);
m3 = abs(y3); % magnitud
% SEÑAL 4-------------------------------------------------------------
y4 = fft(x4);
M4 = ABS(Y4); % MAGNITUD
```

Y su fase, para ello se eliminan las frecuencias muy pequeñas, que pueden considerarse ruido, están se transforman en 0, aquellas que son menores a 1e-6, después se calcula el ángulo de la transformada en radianes.

```
% Señal 1-------------------------------------------------------------
y1(m1<1e-6) = 0;
p1 = unwrap(angle(y1)); % fase
% SEÑAL 2-------------------------------------------------------------
Y2(M2<1E-6) = 0;
p2 = unwrap(angle(y2)); % fase
% SEÑAL 3-------------------------------------------------------------
Y3(M3<1E-6) = 0;
p3 = unwrap(angle(y3)); % fase
% SEÑAL 4-------------------------------------------------------------
Y4(M4<1E-6) = 0;
p4 = unwrap(angle(y4)); % fase
```

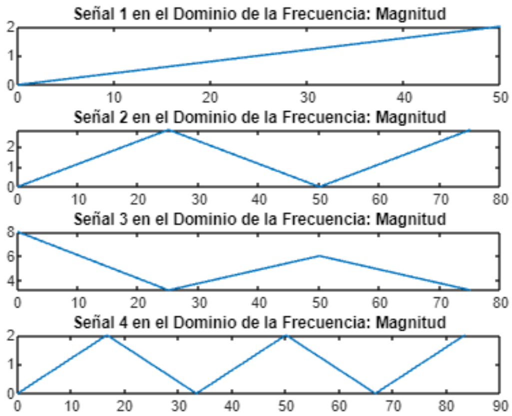
Y se genera un vector de la nueva longitud de “muestras”, que serán las frecuencias resultantes de la transformada, esto hace que el vector de muestras en el tiempo se ajuste al vector de muestras en la frecuencia y se pueda graficar.

```
f1 = (0:length(y1)-1)*100/length(y1);
F2 = (0:LENGTH(Y2)-1)*100/LENGTH(Y2);
F3 = (0:LENGTH(Y3)-1)*100/LENGTH(Y3);
f4 = (0:length(y4)-1)*100/length(y4);
```

Se grafican los vectores de magnitud de todas las señales.

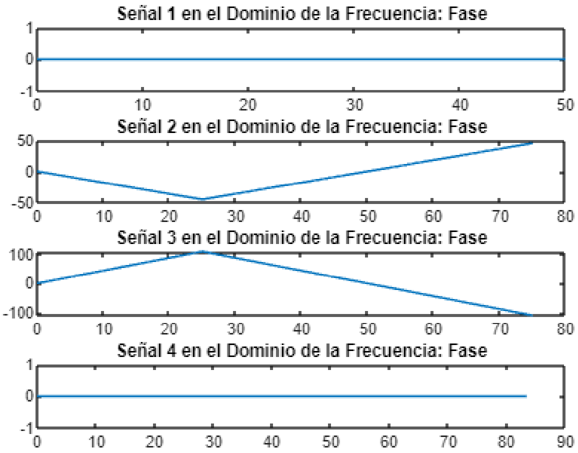
```
% SEÑAL 1-------------------------------------------------------------
subplot(4,1,1)
plot(f1,m1)
title("señal " + num2str(1) + " en el dominio de la frecuencia: magnitud")

% Señal 2-------------------------------------------------------------
subplot(4,1,2)
plot(f2,m2)
title("señal " + num2str(2) + " en el dominio de la frecuencia: magnitud")

% Señal 3-------------------------------------------------------------
subplot(4,1,3)
plot(f3,m3)
title("señal " + num2str(3) + " en el dominio de la frecuencia: magnitud")
```

```
% Señal 4---------------------------------------------------------------
subplot(4,1,4)
plot(f4,m4)
title("señal " + num2str(4) + " en el dominio de la frecuencia: magnitud")
```



Y se grafican sus vectores de fase.

```
% SEÑAL 1---------------------------------------------------------------
subplot(4,1,1)
plot(f1,p1*180/pi)
title("señal " + num2str(1) + " en el dominio de la frecuencia: fase")

% Señal 2---------------------------------------------------------------
subplot(4,1,2)
plot(f2,p2*180/pi)
title("señal " + num2str(2) + " en el dominio de la frecuencia: fase")

% Señal 3---------------------------------------------------------------
subplot(4,1,3)
plot(f3,p3*180/pi)
title("señal " + num2str(3) + " en el dominio de la frecuencia: fase")

% Señal 4---------------------------------------------------------------
subplot(4,1,4)
plot(f4,p4*180/pi)
title("señal " + num2str(4) + " en el dominio de la frecuencia: fase")
```



COMO SE PUDE OBSERVAR EN LAS GRÁFICAS SE MUESTRA LA AMPLITUD DE LOS COMPONENTES DE LA FRECUENCIA, ESTO EN CUANTO A LA MAGNITUD, Y POR SU PARTE EN DONDE INICIA Y TERMINA LA CONTIBUCIÓN DE LAS FRECUENCIAS. ESTO ES

Paso 2. Dada la secuencia de longitud N=4, definida por $\boldsymbol{x}[\boldsymbol{n}] = \mathbf{cos}\left(\frac{\pi}{2}\boldsymbol{n}\right)$ para n= 0,1,2,3 calcula su Transformada Discreta de Fourier (DFT).
Primero se escribe el vector de muestras para evaluar la función:

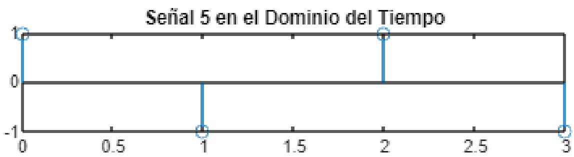
```
% VECTOR DE TIEMPO
T = [0,1,2,3];
```

Y se escribe la señal coseno:

```
X5 = COS(PI*T);
```

Se grafica utilizando la función stem:

```
subplot(3,1,1)
stem(t,x5)
title("Señal " + num2str(5) + " en el dominio del tiempo")
```



Al igual que el paso anterior se calcula su transformada utilizando la función de matlab, que también puede recibir una función, como la definida anteriormente, en vez de una secuencia.

```
Y5 = FFT(X5);
```

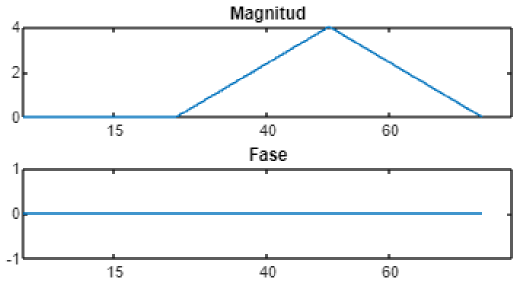
Asimismo, se calcula su magnitud y su fase tal como en el paso anterior.

```
M5 = ABS(Y5); % MAGNITUD
y5(m5<1e-6) = 0;
p5 = unwrap(angle(y5)); % fase
f5 = (0:length(y5)-1)*100/length(y5);
```

Y las graficamos:

```
SUBPLOT(3,1,2)
plot(f5,m5)
title('magnitud de la señal coseno')
```

```
ax = gca;
ax.xtick = [15 40 60 85];

subplot(3,1,3)
plot(f,p*180/pi)
title('fase de la señal coseno')
ax = gca;
ax.xtick = [15 40 60 85];
```



Paso 3. Calcula la Transformada Discreta Inversa de Fourier (IDFT) de las secuencias:

A) $\boldsymbol{X} = (\mathbf{0}, \mathbf{2})$
B) $\boldsymbol{X} = (\mathbf{0}, -\mathbf{2}\boldsymbol{j}, \mathbf{0}, \mathbf{2}\boldsymbol{j}\ )$
C) $\boldsymbol{X} = (\mathbf{0}, \mathbf{4}, \mathbf{0}, \mathbf{0}\ , \mathbf{4})$

Ahora se realiza el proceso inverso, utilizando la función de Matlab “ifft” para calcular la transformada de Fourier Inversa, que recibe una secuencia que representa la transformada de Fourier de una señal. Primero se definen las secuencias de frecuencias en forma de vectores:

```
Y1=[0,2];
Y2=[0,-2J,0,2J];
Y3=[0,4,0,0,4];
```

Después, se calcula la transformada por medio de la función, lo que devuelve la transformada inversa, es decir una secuencia en el tiempo:

```
X1 = IFFT(Y1);
X2 = IFFT(Y2);
X3 = IFFT(Y3);
```

Para observar los resultados de la transformación se imprimen los vectores de las secuencias resultantes:

Para la señal 1:

```
IFFT DE SEÑAL 1
     1    -1
```

Para la señal 2:

```
IFFT DE SEÑAL 2
     0     1     0    -1
```

Para la señal 3:

```
IFFT DE SEÑAL 3
    1.6000    0.4944   -1.2944   -1.2944    0.4944
```

PASO 3. COMPLETE LA DEFINICIÓN DE LA FUNCIÓN *DTFT* QUE ACEPTA LAS SIGUIENTES ENTRADAS:
F → UNA FUNCIÓN ANÓNIMA DE TIEMPO CONTINUO QUE SE VA A MUESTREAR, esta es una expresión matemática cualquiera que describe una señal continua, por ejemplo, la siguiente señal sinusoidal:

```
% SEÑAL
F = @(t) sin(2*pi*5*t);
```

T → UN PERÍODO DE MUESTREO, esto es el intervalo de tiempo entre cada muestra.

```
% PERIODO DE MUESTREO
t = 0.01;
```

N → EL NÚMERO DE MUESTRAS QUE SE INCLUIRÁN A CADA LADO DE 0 EN LA SUMA DTFT (ES DECIR, LAS MUESTRAS REALMENTE ESTÁN INVOLUCRADAS EN LA SUMA), esto quiere decir que se creará un vector de tiempo con el número de muestras (-N,N)

```
% NÚMERO DE MUESTRAS
N = 100;
```

W → UN VECTOR DE FRECUENCIAS EN LAS QUE MUESTREAR EL DTFT, esto es un vector de frecuencias a las que se quiere muestrear, es decir que se realizará una interpolación a estas muestras.

```
% VECTOR DE FRECUENCIAS EN LOS QUE SE MUESTREARÁ LA DTFT
W = linspace(-pi, pi, n);
```

Y DEVUELVE LA SALIDA:
F → UN VECTOR COMPLEJO CORRESPONDIENTE A LA DTFT MUESTREADA EN LOS PUNTOS EN W, este vector de salida representa los valores que resultan de la Transformada de Fourier.

A CONTINUACIÓN, SE DESCRIBE LA FUNCIÓN DE MATLAB DESARROLLADA A PARTIR DE LA DESCRIPCI´N DE LOS PARÁMETROS QUE RECIBE LA FUNCIÓN.
La función primero crea el vector de tiempo de -N a N, y lo guarda como el arreglo t, después evalúa cada una de esas muestras de tiempo en la función anónima pasada como parámetro, y cada muestra se multiplica por el periodo de muestreo. Se calcula la DFT de esa evaluación, utilizando la función integrada de Matlab “fft”.
Posteriormente se hace la interpolación a las frecuencias en W, utilizando la función interp1 de Matlab.
Finalmente, se calcula la magnitud y la fase de la transformada, como se realizó en los pasos anteriores, esto únicamente con el fin de poder graficar la transformada.

LA FUNCIÓN ES LA SIGUIENTE:

```
FUNCTION [F_DFT, M, P, FREC] = DTFT(F, T, N, W)
    % vector de tiempo discreto
    t = (-N:N);

    % evaluar muestras en tiempo discreto
    x = F(t * T);

    % calcular dft
    Y = fft(x);

    % interpolar la dft en las frecuencias deseadas (w)
    f_dft = interp1(t, Y, W, 'spline');

    % calcular la magnitud y fase de la transformada para graficarla
    M = abs(Y); % magnitud
    Y(M<1e-6) = 0;
    P = unwrap(angle(Y)); % fase
    frec = (0:length(Y)-1)*100/length(Y);
```

```
end
```

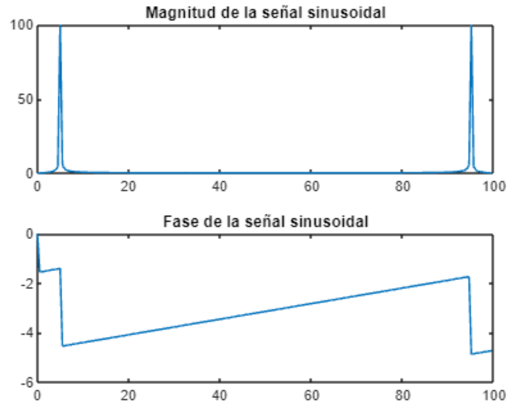
Y UN EJEMPLO DE APLICACIÓN ES:

```
% SEÑAL
F = @(t) sin(2*pi*5*t);

% periodo de muestreo
T = 0.01;

% número de muestras
N = 100;

% vector de frecuencias en los que se muestreará la dtft
W = linspace(-pi, pi, N);

% llamar a la funcion
[F_dft, M, P, frec] = DTFT(F, T, N, W);
fprintf('Función de dtft');
disp(F_dft);

% graficar la transformada de la señal
subplot(2,1,1)
stem(frec, M);
title('Magnitud de la señal sinusoidal')
subplot(2,1,2)
stem(frec, P);
title('Fase de la señal sinusoidal')
```

LOS RESULTADOS DE LA TRANSOFRMADA DE LA SEÑAL PARA EL EJEMPLO ANTERIOR SON LOS SIGUIENTES:
VECTOR DE SALIDA.

```
FUNCIÓN DE DTFT
   -0.1584 + 0.0090I  -0.1584 + 0.0089I  -0.1584 + 0.0087I  -0.1584 + 0.0086I  -0.1584 +
   0.0084I  -0.1584 + 0.0082I  -0.1584 + 0.0081I  -0.1584 + 0.0079I  -0.1584 + 0.0078I
   -0.1584 + 0.0076I  -0.1584 + 0.0074I  -0.1584 + 0.0073I  -0.1584 + 0.0071I  -0.1584 +
   0.0070I  -0.1584 + 0.0068I  -0.1584 + 0.0067I  -0.1584 + 0.0065I  -0.1584 + 0.0063I
   -0.1584 + 0.0062I  -0.1584 + 0.0060I  -0.1584 + 0.0059I  -0.1584 + 0.0057I  -0.1584 +
   0.0056I  -0.1584 + 0.0054I  -0.1584 + 0.0052I  -0.1584 + 0.0051I  -0.1584 + 0.0049I
   -0.1584 + 0.0048I  -0.1584 + 0.0046I  -0.1584 + 0.0045I  -0.1584 + 0.0043I  -0.1584 +
   0.0041I  -0.1584 + 0.0040I  -0.1584 + 0.0038I  -0.1584 + 0.0037I  -0.1584 + 0.0035I
   -0.1584 + 0.0034I  -0.1584 + 0.0032I  -0.1584 + 0.0030I  -0.1584 + 0.0029I  -0.1584 +
   0.0027I  -0.1584 + 0.0026I  -0.1584 + 0.0024I  -0.1584 + 0.0023I  -0.1584 + 0.0021I
   -0.1584 + 0.0019I  -0.1584 + 0.0018I  -0.1584 + 0.0016I  -0.1584 + 0.0015I  -0.1584 +
   0.0013I  -0.1584 + 0.0012I  -0.1584 + 0.0010I  -0.1584 + 0.0008I  -0.1584 + 0.0007I
   -0.1584 + 0.0005I  -0.1584 + 0.0004I  -0.1584 + 0.0002I  -0.1584 + 0.0001I  -0.1584 -
   0.0001I  -0.1584 - 0.0003I  -0.1584 - 0.0004I  -0.1584 - 0.0006I  -0.1584 - 0.0007I
   -0.1584 - 0.0009I  -0.1584 - 0.0010I  -0.1584 - 0.0012I  -0.1584 - 0.0014I  -0.1584 -
   0.0015I  -0.1584 - 0.0017I  -0.1584 - 0.0018I  -0.1584 - 0.0020I  -0.1584 - 0.0021I
   -0.1584 - 0.0023I  -0.1584 - 0.0025I  -0.1584 - 0.0026I  -0.1584 - 0.0028I  -0.1584 -
   0.0029I  -0.1584 - 0.0031I  -0.1584 - 0.0032I  -0.1584 - 0.0034I  -0.1584 - 0.0036I
   -0.1584 - 0.0037I  -0.1584 - 0.0039I  -0.1584 - 0.0040I  -0.1584 - 0.0042I  -0.1584 -
   0.0043I  -0.1584 - 0.0045I  -0.1584 - 0.0047I  -0.1584 - 0.0048I  -0.1584 - 0.0050I
   -0.1584 - 0.0051I  -0.1584 - 0.0053I  -0.1584 - 0.0054I  -0.1584 - 0.0056I  -0.1584 -
   0.0058I  -0.1584 - 0.0059I  -0.1584 - 0.0061I  -0.1584 - 0.0062I  -0.1584 - 0.0064I
   -0.1584 - 0.0065I
```

GRÁFICAS DE LA TRANSFORMADA DE SEÑAL.



CONCLUSIÓN:
A LO LARGO DE ESTA PRÁCTICA SE ANALIZARON DIFERENTES CONCEPTOS RELACIONADOS CON LA TRANSFORMADA DE FOURIER DISCRETA, Y SU REPRESENTACIÓN EN EL DOMINIO DE LA FRECUENCIA. SE UTILIZÓ LA HERRAMIENTA MATLAB QUE YA TIENE INTEGRADAS FUNCIONES QUE FACILITAN LOS CÁLCULOS DE LAS TRANSOFORMADAS, CON ALGORITMOS OPTIMIZADOS PARA QUE TODAS LAS OPERACIONES SEAN MÁS RÁPIDAS Y SE PUEDA CALCULAR LA DFT DE SEÑALES MÁS GRANDES. ADEMÁS, SE PUDO ANALIZAR GRÁFICAMENTE LO QUE OCURRE AL CALCULAR LA TRANSFORMADA DE UNA SEÑAL, CALCULANDO LA MAGNITUD Y LA FASE, PARA IDENTIFICAR ESTAS FRECUENCIAS. FINALMENTE, SE PUEDE DECIR QUE ES IMPORTANTE PODER ANALIZAR SEÑALES DESDE EL DOMINIO DE LA FRECUENCIA, YA QUE ES UTILIZADA PARA MUCHAS APLICACIONES, YA SEA EN TELECOMUNICACIONES, PROCESAMIENTO DE AUDIO, EN MÚSICA, PROCESAMIENTO DE IMÁGENES, ENTRE OTRAS.

FECHA FINAL DE ENTREGA: lunes, 30 de octubre de 2023