6.300 Final Exam

Spring 2023

Name: **Answers**

Kerberos/Athena Username:

7 questions 3 hours

- Please **write your name and your Athena username in the box above**, and please do not write your name on any of the other pages of the exam.
- Please **WAIT** until we tell you to begin.
- This quiz is closed-book, but you may use three 8.5×11 sheets of paper (both sides) as a reference.
- You may **NOT** use any electronic devices (including computers, calculators, phones, etc.).
- If you have questions, please **come to us at the front** to ask them.
- Enter all answers in the boxes provided. Work on other pages with QR codes may be taken into account when assigning partial credit. **Please do not write on the QR codes.**
- If you finish the exam more than 10 minutes before the end time, please quietly bring your exam to us at the front of the room. If you finish within 10 minutes of the end time, please remain seated so as not to disturb those who are still finishing their quizzes.
- You may not discuss the details of the quiz with anyone other than course staff until final quiz grades have been assigned and released.

1 Transforms

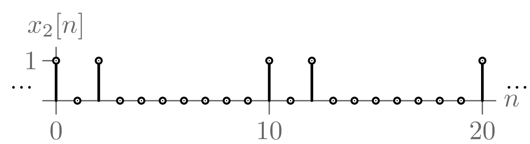
Find the Fourier series coefficients of the signal $x_1(\cdot)$, analyzed with T chosen to be the fundamental period of $x_1(\cdot)$:

$$x_1(t) = 2\cos\left(\frac{\pi}{2}t\right) + 4\cos\left(\frac{\pi}{3}t\right)$$

In the box below, write a simple, closed-form answer for $X_1[k]$:

$X_1[k] = 2\delta[k-2] + 2\delta[k+2] + \delta[k-3] + \delta[k+3]$

Find the Fourier series coefficients of the signal $x_2[\cdot]$, shown below, which is periodic in $N = 10$.



In the box below, write a simple, closed-form answer for $X_2[k]$:

$X_2[k] = \frac{1}{10}\left(1 + e^{-j\frac{2\pi k}{5}}\right)$

Find the Fourier transform of the signal $x_3(\cdot)$ as defined below:

$$x_3(t) = \begin{cases} -3 & \text{if } -1 \le t \le 2 \\ 0 & \text{otherwise} \end{cases}$$

In the box below, write a simple, closed-form answer for $X_3(\omega)$:

$$X_3(\omega) = e^{-j\frac{\omega}{2}} \left(\frac{-6 \sin\left(\frac{3}{2}\omega\right)}{\omega} \right)$$

Find the Fourier transform of the signal $x_4[\cdot]$, defined below:

$$x_4[n] = \delta[n+3] + \delta[n+1] - \delta[n-1] + \delta[n-3]$$

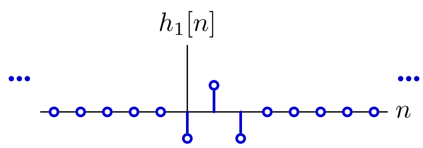
In the box below, write a simple, closed-form answer for $X_4(\Omega)$:

$$X_4(\Omega) = 2\cos(3\Omega) + 2j\sin(\Omega)$$

2 Filtering

2.1 Frequency Response

Sketch the magnitude and phase of the frequency response of a linear, time-invariant system with the following unit-sample response:

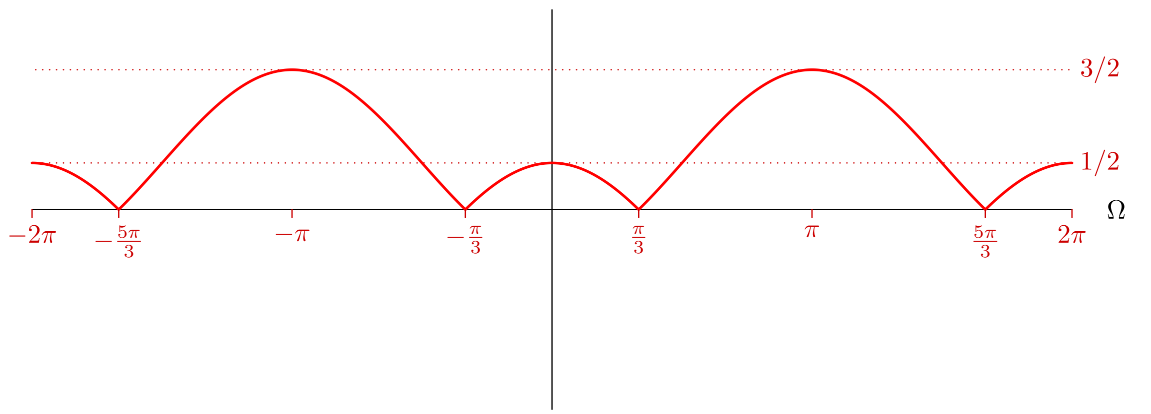


$$h_1[n] = -\frac{1}{2}\delta[n] + \frac{1}{2}\delta[n-1] - \frac{1}{2}\delta[n-2]$$

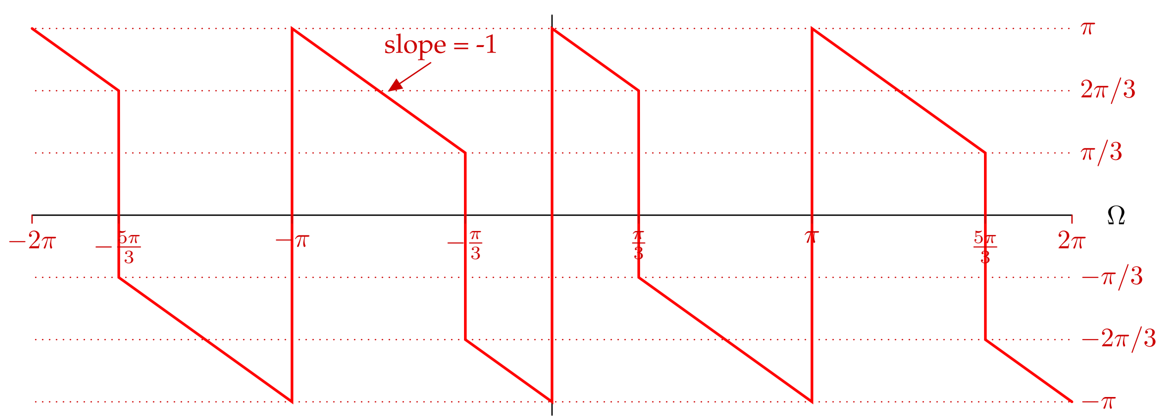
Label all important magnitudes, angles, and frequencies in your sketches.

This signal is a shifted version of a symmetric signal. We can use the time-shift property to find: $H_1(\Omega) = e^{-j\Omega}\left(\frac{1}{2} - \cos(\Omega)\right)$

magnitude



phase



2.2 Outputs

Consider a different LTI system with a unit sample response $h_2[\cdot]$ given by $h_2[n] = \delta[n] + \delta[n-2]$

For each of the input signals $x_i[\cdot]$ below, assume that the response of the system to that input is given by $y_i[\cdot]$. For each, is it possible to represent $y_i[n]$ as a single pure sinusoid of the form $A_i \cos(\Omega_i n + \phi_i)$? If so, specify the appropriate values of A_i, Ω_i, and ϕ_i by entering a single number in each box (square roots, π, and fractions are all OK). Otherwise, write "`none`" in all three boxes. If a box is irrelevant, write "`any`" in that box.

A good place to start here is to find the frequency response of this system. Using linearity and the time-shift property (or just direct application of the analysis formula), we find: $H(\Omega) = e^{-j\Omega} 2\cos(\Omega)$

Knowing this, we know that if the input to the system is a cosine $x[n] = \cos(\Omega_1 n + \phi_1)$, then the output will be a cosine at the same frequency, scaled and shifted according to the frequency response:
$y[n] = |H(\Omega_1)| \cos(\Omega_1 n + \phi_1 + \angle(H(\Omega_1)))$

(note that there are multiple reasonable approaches and multiple possible answers; the ones shown here are the ones that follow most directly from the idea above)

If $x_1[n] = 4\delta[n]$, is $y_1[n]$ expressible as $y_1[n] = A_1 \cos(\Omega_1 n + \phi_1)$?

$A_1 =$ `none` $\Omega_1 =$ `none` $\phi_1 =$ `none`

If $x_2[n] = 3$, is $y_2[n]$ expressible as $y_2[n] = A_2 \cos(\Omega_2 n + \phi_2)$?

$A_2 = 6$ $\Omega_2 = 0$ $\phi_2 = 0$

If $x_3[n] = \cos(\pi n/3)$, is $y_3[n]$ expressible as $y_3[n] = A_3 \cos(\Omega_3 n + \phi_3)$?

$A_3 = 1$ $\Omega_3 = \pi/3$ $\phi_3 = -\pi/3$

If $x_4[n] = \sin(\pi n/3)$, is $y_4[n]$ expressible as $y_4[n] = A_4 \cos(\Omega_4 n + \phi_4)$?

$A_4 = 1$ $\Omega_4 = \pi/3$ $\phi_4 = -5\pi/6$

If $x_5[n] = \cos(4\pi n/3)$, is $y_5[n]$ expressible as $y_5[n] = A_5 \cos(\Omega_5 n + \phi_5)$?

$A_5 = 1$ $\Omega_5 = 4\pi/3$ $\phi_5 = -\pi/3$

If $x_6[n] = \sin(\pi n/2) + \cos(\pi n/2)$, is $y_6[n]$ expressible as $y_6[n] = A_6 \cos(\Omega_6 n + \phi_6)$?

$A_6 = 0$ $\Omega_6 =$ `any` $\phi_6 =$ `any`

If $x_7[n] = (-1)^n$, is $y_7[n]$ expressible as $y_7[n] = A_7 \cos(\Omega_7 n + \phi_7)$?

$A_7 = 2$ $\Omega_7 = \pi$ $\phi_7 = 0$

3 DT Transforms

3.1 Transforms

Let $X_1[k]$ represent the DFT of the signal when analyzed with $N = 8$:

$$x_1[n] = \begin{cases} 1 & \text{if } n \in \{0, 1, 7\} \\ 2 & \text{if } n = 2 \\ -2 & \text{if } n = 6 \\ 0 & \text{otherwise} \end{cases}$$

Determine a closed form expression for $X_1[k]$ in terms of sin and cos functions (no complex exponentials), and enter your expression in the box below.

$$X_1[k] = \frac{1}{8}\left(1 + 2\cos\left(\frac{\pi}{4}k\right) - 4j\sin\left(\frac{\pi}{2}k\right)\right)$$

Let $X_2[k]$ represent the DFT of the signal analyzed with $N = 16$:

$$x_2[n] = (-j)^n + e^{j\pi n/4}$$

Write a closed form expression for $X_2[k]$ in the box below. Your answer should not contain any complex exponentials.

$$X_2[k] = X_2[k+16] = \delta[k-2] + \delta[k-12]$$

3.2 2D Basis Functions

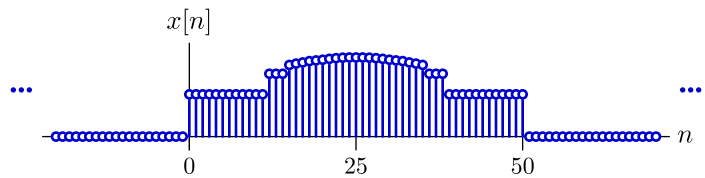
Let $F_i[k_r, k_c]$ represent the Discrete Fourier Transforms (DFTs) of the following two-dimensional signals $f_i[r, c]$, where $0 \le r < 64$ and $0 \le c < 64$. For each signal, list all of the indices of the form $[k_r, k_c]$, where $-31 \le k_r \le 32$ and $-31 \le k_c \le 32$, for which $F_i[k_r, k_c] \ne 0$ in the box provided.

Notice that $f_i[r, c]$ may have positive, negative, or even complex values.

$f_i[r, c]$	list of 2D indices ($[k_r, k_c]$) for which $F_i[k_r, k_c] \ne 0$
$\left(\cos\frac{6\pi r}{64}\right)\left(\cos\frac{8\pi c}{64}\right)$	$[3, 4], [3, -4], [-3, 4], [-3, -4]$
$\left(\cos\frac{6\pi r}{64}\right) + \left(\cos\frac{8\pi c}{64}\right)$	$[3, 0], [-3, 0], [0, 4], [0, -4]$
$\cos\left(\frac{6\pi r}{64} + \frac{8\pi c}{64}\right)$	$[3, 4], [-3, -4]$
$\cos\left(\frac{6\pi r}{64} - \frac{8\pi c}{64}\right)$	$[3, -4], [-3, 4]$
$\sin\left(\frac{6\pi r}{64} + \frac{8\pi c}{64}\right)$	$[3, 4], [-3, -4]$
$\left(1+\cos\frac{6\pi r}{64}\right)\left(1+\cos\frac{8\pi c}{64}\right)$	$[0, 0], [3, 0], [-3, 0], [0, 4], [0, -4], [3, 4], [3, -4], [-3, 4], [-3, -4]$
$e^{j2\pi\frac{3r}{64}}e^{j2\pi\frac{4c}{64}}$	$[3, 4]$
$e^{j2\pi\frac{3r}{64}} + e^{j2\pi\frac{4c}{64}}$	$[3, 0], [0, 4]$
$(-1)^r + (-1)^c$	$[32, 0], [0, 32]$
$\left(\cos\frac{32\pi r}{64}\right)^2$	$[0, 0], [32, 0]$

4 Return of the Dome

Consider the following signal, which we'll refer to as $x[n]$:



In addition, consider two additional signals (note that H_1 is specified by its DTFT, whereas h_2 is specified in the time domain):

$$H_1(\Omega) = j\sin(5\Omega)$$

$$h_2[n] = \sin\left(\frac{10\pi n}{50}\right)$$

For each of the signals below, indicate which of the graphs on the facing page (page 9) whose shape matches the signal most closely by writing a single letter in each box. Note that the graphs are not necessarily represented on the same vertical scale.

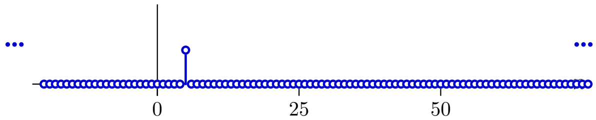
$(x \times h_1)[n] =$ B

$(x \times h_2)[n] =$ I

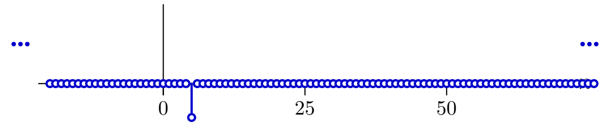
$(x * h_1)[n] =$ M

$(x * h_2)[n] =$ L

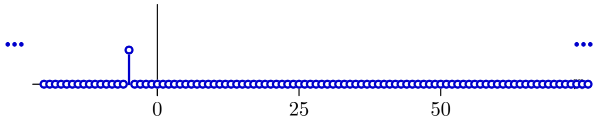
Graph A



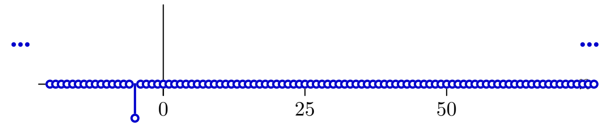
Graph B



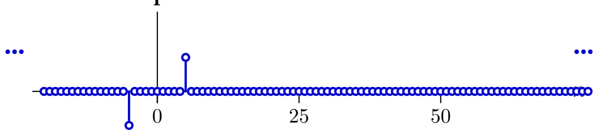
Graph C



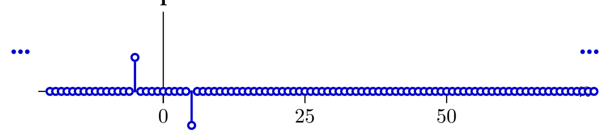
Graph D



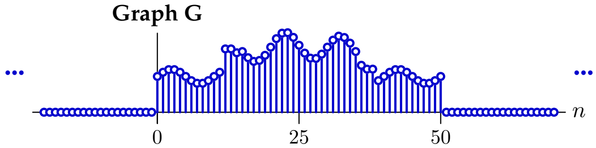
Graph E



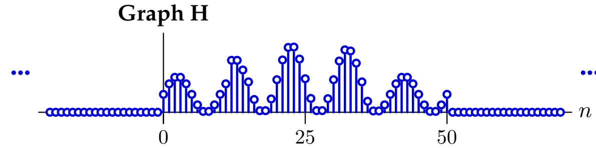
Graph F



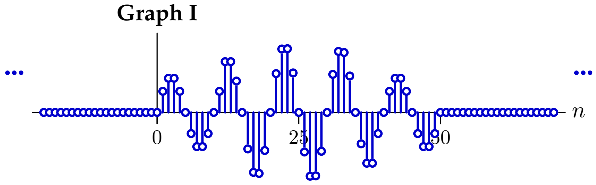
Graph G



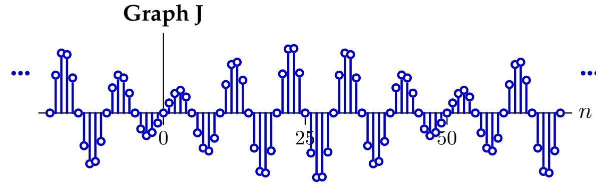
Graph H



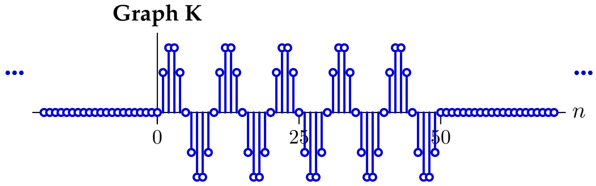
Graph I



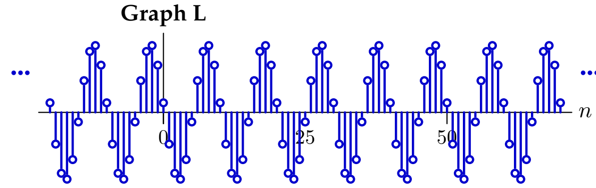
Graph J



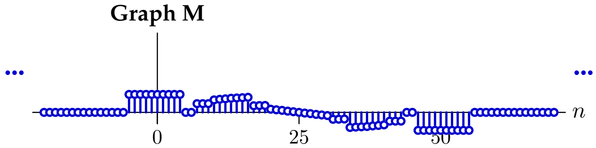
Graph K



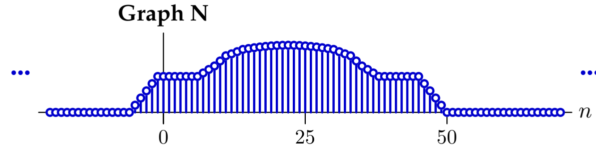
Graph L



Graph M



Graph N



5 DT Calculus

Part 1: Derivatives

A crude approximation to differentiation on a discrete signal is a *first difference*:

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

While there may be better means of approximating derivatives, this method is easy to implement and can work reasonably well in some scenarios to get a sense of how a DT function is changing with time.

The following code implements a first difference of this form. Its input is a Python list representing one period of x, i.e., $[x[0], x[1], x[2], \ldots, x[N-2], x[N-1]]$. It returns a list of the same length representing a single period of the derivative of that signal (using the first difference as described above).

```
def first_difference_time(x):
    out = []
    for n in range(len(x)):
        out.append(x[n] - x[n-1])
    return out
```

We would like to consider implementing a function to perform this same operation but by operating in the frequency domain. Consider the function below, which is designed to take as input a Python list containing the DTFS coefficients $X[k]$ of x, i.e., $[X[0], X[1], X[2], \ldots, X[N-2], X[N-1]]$ and to return a list of the same length containing the DTFS coefficients of the derivative of x (approximated using the first difference as described above).

```
01 |   def first_difference_frequency(X):
02 |       out = []
03 |       for k in range(len(X)):
04 |           A = # YOUR CODE HERE
05 |           out.append(A * X[k])  # multiply X[k] by A
06 |       return out
```

Most of the function has been filled in for you, except for the definition of `A` within each iteration of the loop. Is it possible to insert code where indicated on line 4 to make this function correctly compute the DTFS coefficients of the first difference of x, without changing the other parts of the function? You may assume that familiar constants/functions (`pi`, `cos`, `sin`, `e`, etc.) from the `math` module are available to you.

Circle One: Yes / No

If yes, write the necessary code for setting `A`. If not, explain briefly.

```
A = 1 - e ** (-2j * pi * k / len(X))
```

Part 2: Integrals

A crude approximation to integration on a discrete signal is a *running sum*:

$$y[n] = \sum_{i=-\infty}^{n} x[i]$$

Note that this signal is only finite-valued and periodic if $X[0] = 0$, i.e., if the original signal has no DC offset. We will make that assumption moving forward.

The following code implements a running sumof this form. Its input is a Python list representing one period of x, i.e., $[x[0], x[1], x[2], \dots, x[N-2], x[N-1]]$. It returns a list of the same length representing a single period of the integral of that signal (using the running sum as described above).

```
01 |   def running_sum_time(x):
02 |       out = []
03 |       for n in range(len(x)):
04 |           out.append(sum(x[:n+1]))
05 |       return out
```

We would like to consider implementing a function to perform this same operation but by operating in the frequency domain. Consider the function below, which is designed to take as input a Python list containing the DTFS coefficients $X[k]$ of x, i.e., $[X[0], X[1], X[2], \dots, X[N-2], X[N-1]]$ and to return a list of the same length containing the DTFS coefficients of the integral of x (approximated using the running sum as described above).

```
01 |   def running_sum_frequency(X):
02 |       out = []
03 |       for k in range(len(X)):
04 |           A = # YOUR CODE HERE
05 |           out.append(A * X[k])  # multiply X[k] by A
06 |       return out
```

Most of the function has been filled in for you, except for the definition of `A` within each iteration of the loop. Is it possible to insert code where indicated on line 4 to make this function correctly compute the DTFS coefficients of the running sum of x, without changing the other parts of the function? You may assume that familiar constants/functions (`pi`, `cos`, `sin`, `e`, etc.) from the `math` module are available to you.

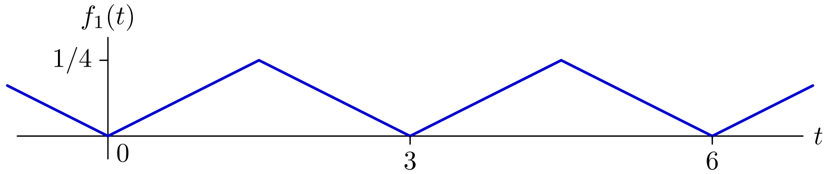
Circle One: Yes / No

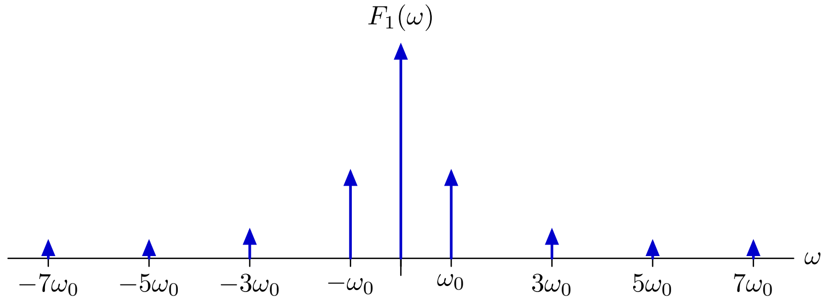
If yes, write the necessary code for setting `A`. If not, explain briefly.

```
A = 1 / (1 - e ** (-2j * pi * k / len(X)))
```

6 Related Transforms

A signal $f_1(t)$, which is periodic in $T = 3$, is shown below, as is a portion of its (purely real) Fourier transform.





6.1 ω Values

Note that the Fourier transform of this signal is zero for most values of ω! It is only nonzero at integer multiples of some value ω_0. Solve for ω_0, and enter your answer in the box below:

$\omega_0 =$ $\frac{2\pi}{3}$

6.2 Second Signal

Consider a second signal, $f_2(t) = -f_1(t-1)$

True or False? $F_2(\omega)$ is nonzero at precisely the same ω values at which $F_1(\omega)$ was nonzero, and nowhere else.

Circle one: **True** / **False** and explain briefly:

Using the time-shift and linearity properties of the CTFT, we can find that $F_2(\omega) = -F_1(-\omega)e^{-j\omega}$.

For every value of ω, $F_2(-\omega)$ will have the same magnitude as $F_1(\omega)$. Since the nonzero ω values come in $+/-$ pairs, everywhere $F_1(\omega)$ was 0, $F_2(\omega)$ will also be 0; and every that $F_1(\omega)$ was nonzero, $F_2(\omega)$ will be nonzero as well.

Is $F_2(\omega)$ is purely real, purely imaginary, or neither?

Circle one: **Purely Real** / **Purely Imaginary** / **Neither** and explain briefly:

$f_2(t)$ is a purely real function of t. Thus, its symmetry (or lack thereof) determines whether its transform will be purely real (iff $f_2(t)$ were a symmetric function of t), purely imaginary (iff $f_2(t)$ were an antisymmetric function of t), or something else.

In this case, after the shift, $f_2(t)$ is neither symmetric nor antisymmetric, so $F_2(\omega)$ is neither purely real nor purely imaginary.

6.3 Third Signal

Consider another signal, $f_3(t) = 4f_1(t/2)$

True or False? $F_3(\omega)$ is nonzero at precisely the same ω values at which $F_1(\omega)$ was nonzero, and nowhere else.

Circle one: **True** / **False** and explain briefly:

Using the linearity and time-scaling properties of the DTFT, we have $F_3(\omega) = F_1(2\omega)$. So while $F_1(\omega)$ was nonzero at $2, \pm\frac{2\pi}{3}, \pm\frac{6\pi}{3}, \pm\frac{10\pi}{3}, \ldots$, we will instead have the $F_3(\omega)$ is nonzero at $0, \pm\frac{\pi}{3}, \pm\frac{3\pi}{3}, \pm\frac{5\pi}{3}, \ldots$

Is $F_3(\omega)$ is purely real, purely imaginary, or neither?

Circle one: **Purely Real** / **Purely Imaginary** / **Neither** and explain briefly:

$f_1(t)$ was a symmetric, purely-real function of t, and the time stretching we're applying here (multiplying the argument by $1/2$) maintains that symmetry; so $f_3(t)$ is a symmetric, purely-real function of t, which means that $F_3(\omega)$ will be a symmetric, purely-real function of ω.

6.4 Fourth Signal

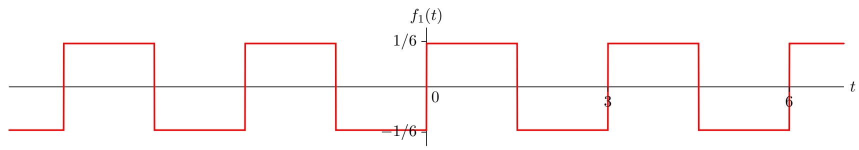
Consider another signal, $f_4(t)$, that is equal to the derivative of $f_1(t)$.

True or False? $F_4(\omega)$ is nonzero at precisely the same ω values at which $F_1(\omega)$ was nonzero, and nowhere else.

Circle one: **True** / **False** and explain briefly:

There are a couple of different ways to see this. One way is to apply the time-derivative property of the CTFT to find that $F_4(\omega) = j\omega F_1(\omega)$. This *almost* results in the same nonzero values, with one notable expection: $F_1(0)$ was nonzero, but $F_4(0) = 0$!

We could also see this by thinking about this function in the time domain. The derivative of this sawtooth wave is a square wave, without the constant offset (the derivative operation zeroes out the constant offset):



Since the DC offset is now 0 but was nonzero, $F_4(\omega)$ and $F_1(\omega)$ are not nonzero at precisely the same ω values.

Is $F_4(\omega)$ is purely real, purely imaginary, or neither?

Circle one: **Purely Real** / **Purely Imaginary** / **Neither** and explain briefly:

We can use either the methods above to think about this problem as well.

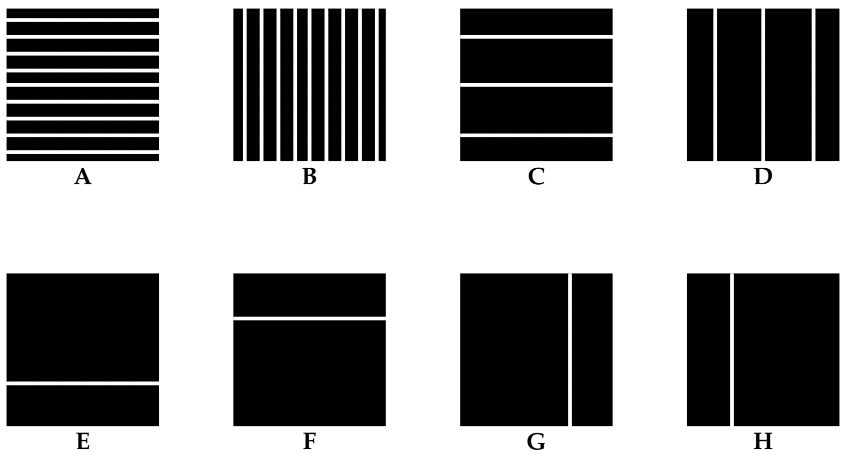
From the frequency-domain view of $F_4(\omega) = j\omega F_1(\omega)$, we can see that because $F_1(\omega)$ was purely real and symmetric, then $F_4(\omega)$ will be purely imaginary (because of the multiplication by j) and antisymmetric (because ω changes signs around 0).

Alternatively, from the time-domain sketch of $f_4(t)$, we can see that in the time domain, this function is purely real and antisymmetric; thus, its transform $F_4(\omega)$ will be purely imaginary and antisymmetric.

7 Shapes

Consider the following 2-D signals (labeled A through H). In each:

- black represents a value of 0, and white represents a value of 1
- the origin $(r = 0, c = 0)$ is in the center of each image
- r increases downward, and c increases to the right
- the image is 63 pixels wide and 63 pixels tall
- all lines are white and one pixel wide



On the last two pages of this exam are 18 plots of DFT coefficient magnitudes.

For each of the combinations of signals below and on the facing page (where $+$ denotes element-wise addition, $\times$ denotes element-wise multiplication, and $\circledast$ denotes circular convolution), indicate which of the numbered images best matches the DFT coefficient magnitudes of that combination. Enter a single number in each box.

$A + B$ matches graph (1-18): 11

$A \times B$ matches graph (1-18): 9

$A \circledast B$ matches graph (1-18): 8

$C + B$ matches graph (1-18): 17

$C \times B$ matches graph (1-18): 18

$C \circledast B$ matches graph (1-18): 8

$A + G$ matches graph (1-18): 10

$A \times G$ matches graph (1-18): 7

$A \circledast G$ matches graph (1-18): 8

$F + G$ matches graph (1-18): 2

$F \times G$ matches graph (1-18): 5

$F \circledast G$ matches graph (1-18): 8

$E \times F$ matches graph (1-18): 14

$E \circledast F$ matches graph (1-18): 13

Worksheet (intentionally blank)

Worksheet (intentionally blank)

Worksheet (intentionally blank)

Worksheet (intentionally blank)

Worksheet (intentionally blank)

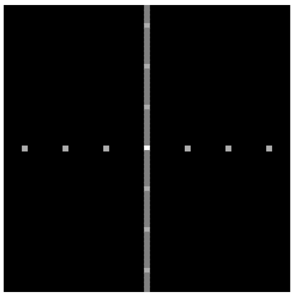
Worksheet (intentionally blank)

Worksheet (intentionally blank)

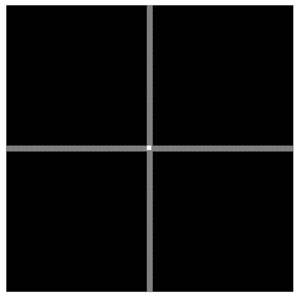
Images for ”Shapes” Problem

Each plot below shows the magnitude of the DFT coefficients associated with some 2-D signal. In each, $(k_r = 0, k_r = 0)$ is in the center of the image. Black represents 0, and pure white represents the highest value in the image (not necessarily 1).

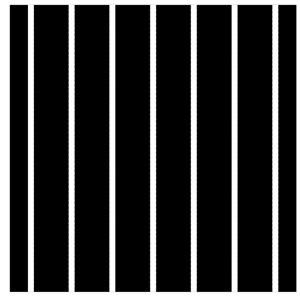
Note that there are nine more images on the following page.



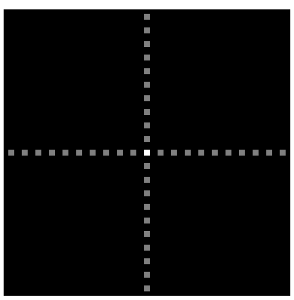
1



2

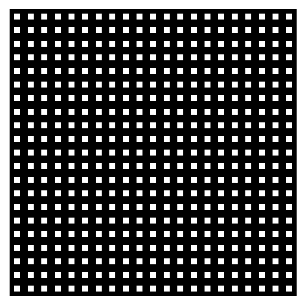


3

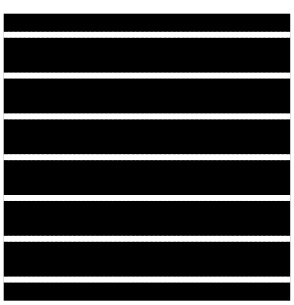


4

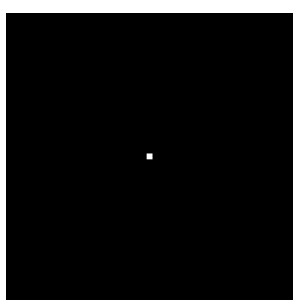
5



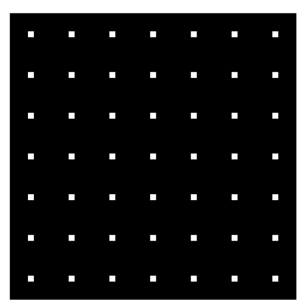
6



7



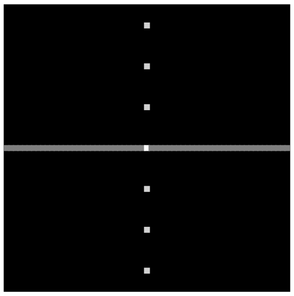
8

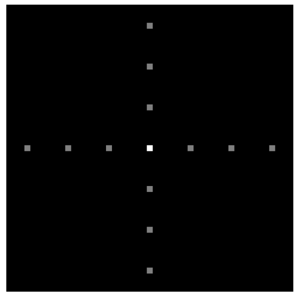


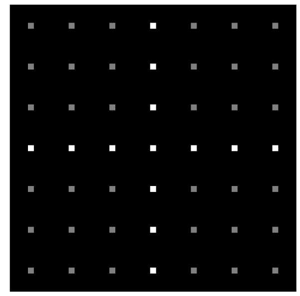
9

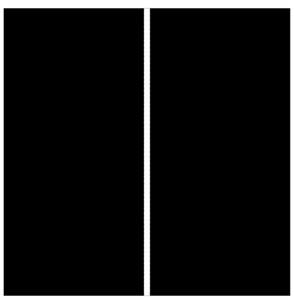
Worksheet (intentionally blank)

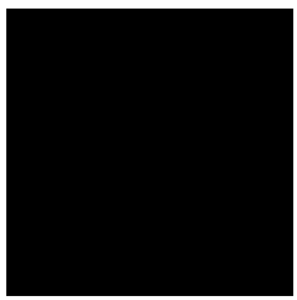
More Images for "Shapes" Problem

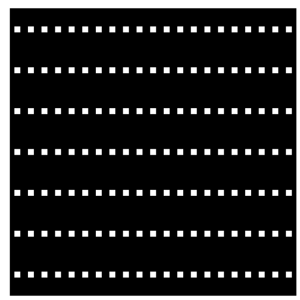

10

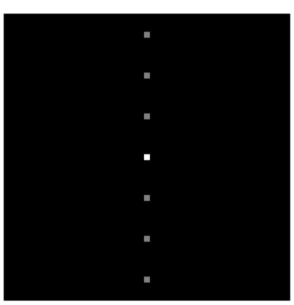

11

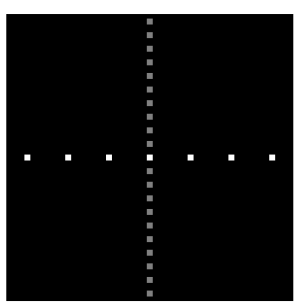

12

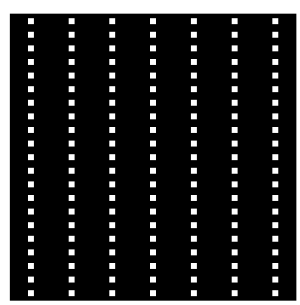

13


14


15


16


17


18

Worksheet (intentionally blank)