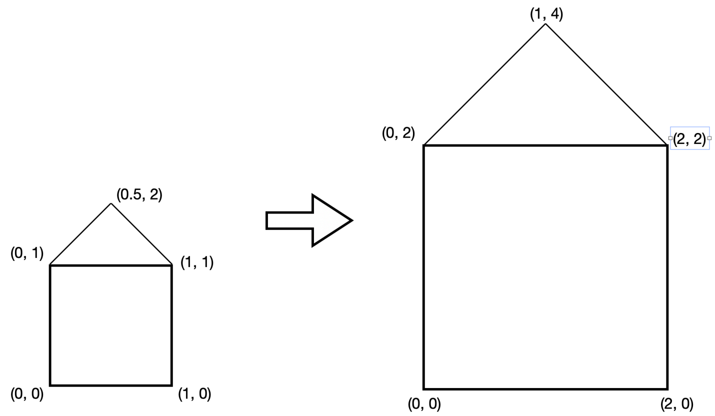
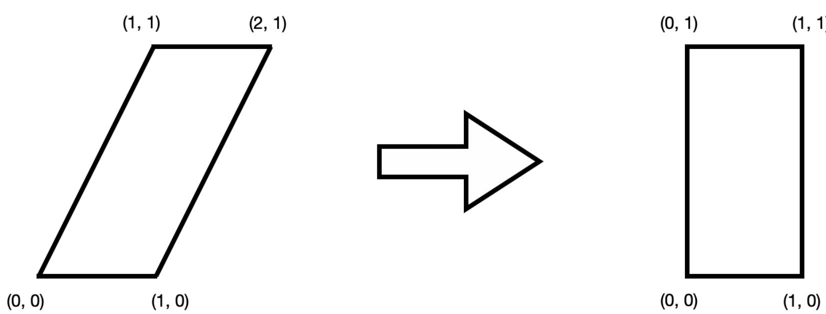
Q1: Linear transformations

(a) Julia has a simple detached house. A life event happens (marriage, child, etc…), and she now needs the house to be **twice as big** (both taller AND wider).



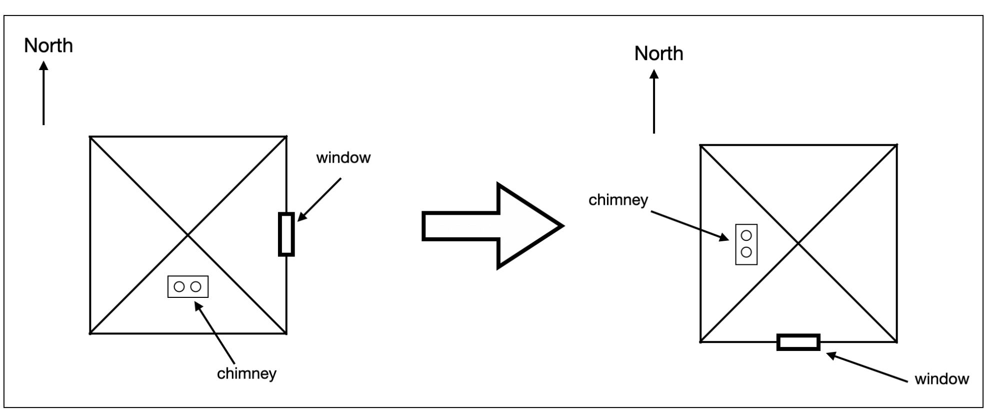
I. Give a matrix which makes her house 2x bigger.
II. What is the name of this transformation?

(b) As she's exploring her new house, she finds that a contractor didn't make a door right. The top of the door is **shifted to the right** and the resulting door frame is diagonal!



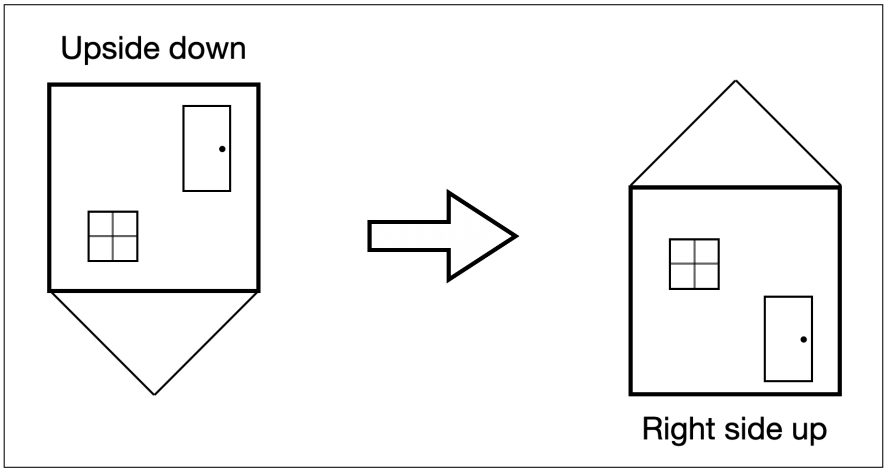
I. Give a matrix which straightens up her door so the frame edges are vertical.
II. What is the name of this transformation?

(c) Julia has a beautiful window, but it faces **east**. She wants it to face **south**.



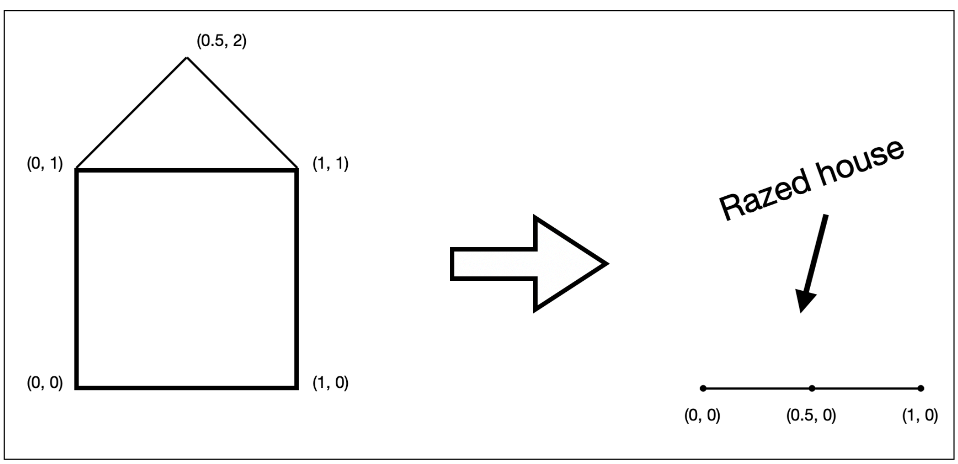
I. Give a matrix which turns her house so the window faces south.
II. What is the name of this transformation?

(e) Julia is leaving Boston for a new life in Australia. She takes his house with him, but discovers that everything in Australia is **upside down**. Oh, no!



I. Give a matrix which flips her house right side up, please.
II. What is the name of this transformation?

(f) Julia decides she needs a new house. She wants to flatten the old one, but keep the footprint.



I. Give a matrix which flattens her house onto the x-axis.
II. What is the name of this transformation?

Q2: Rotations

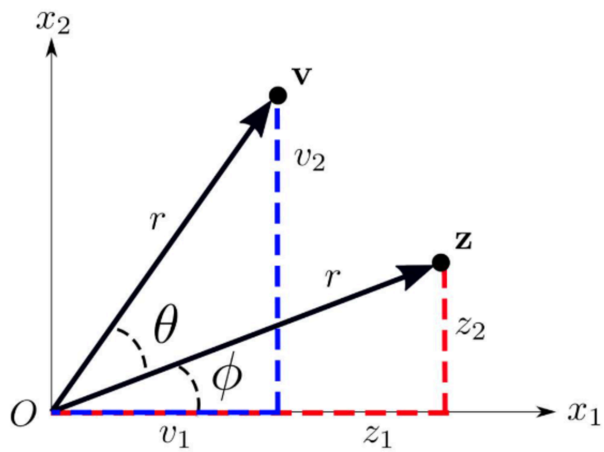
1. Rotations: Intuition and Justification

   Although many of the transformations from the book (Chapter 1.9) can be reasoned intuitively, the rotation matrix is more difficult to justify. Why do these sines and cosines rotate a vector? Let's see!

   Remember that the book defines a rotation by an angle θ as the matrix

$$\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$$

   Consider the following picture of two vectors $\mathbf{v}$ and $\mathbf{z}$, which have the same length r.



$$\mathbf{v} = \begin{bmatrix} v_1 \\ v_2 \end{bmatrix}, \qquad \mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

   (a) Write each of the components of these vectors, v_1, v_2, z_1, z_2, in terms of the angles and lengths θ, ϕ, and r using trigonometry.
   Hint: the inside angle for the "$\mathbf{v}$" triangle is $\phi + \theta$.

(b) For v_1, the horizontal component of **v**, use a trigonometric identity to split up the sum-of-angles inside the $\cos()$.

(c) For v_1, plug in your definitions of z_1 and z_2 to eliminate r and ϕ.

(d) For v_1, factor out the z_1 and z_2 using vector-vector multiplication: put in values for the "..." below.

$$v_1 = \begin{bmatrix} \dots & \dots \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

(e) For v_2, the vertical component of **v**, use a trigonometric identity to split up the sum-of-angles inside the $\sin()$.

(f) For v_2, plug in your definitions of z_1 and z_2 to eliminate r and ϕ.

(g) For v_2, factor out the z_1 and z_2 using vector-vector multiplication: plug in values for the "..." below.

$$v_2 = \begin{bmatrix} \dots & \dots \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix}$$

(h) Finally, using your answers to part (d) and (g), plug in values into a matrix that transforms the vector **z** into the vector **v**. Wow! It's the rotation matrix!

$$\begin{bmatrix} v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} \dots & \dots \\ \dots & \dots \end{bmatrix} \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \qquad \Rightarrow \mathbf{v} = A\mathbf{z}$$

Q3 Homogeneous transformations

1) Background

You learned some basic transformation matrices from Table 1 in Ch. 1.9 of Lay, but in the robotics field, there is this other cool transformation called "homogeneous" transformations that ppl use all the time! A *homogeneous transformation* looks like the below:

$$\mathbf{T} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & t_{forward} \\ \sin(\theta) & \cos(\theta) & t_{left} \\ 0 & 0 & 1 \end{bmatrix}$$

Which is a step (*translation*) and then a turn (*rotation*) in 2D. You can see the **step** is the 2D vector on the far right, and the **turn** is the 2x2 matrix on the top left. (Ignore the bottom **[0 0 1]** row, it's just what makes it "homogeneous".) So here things are broken out:

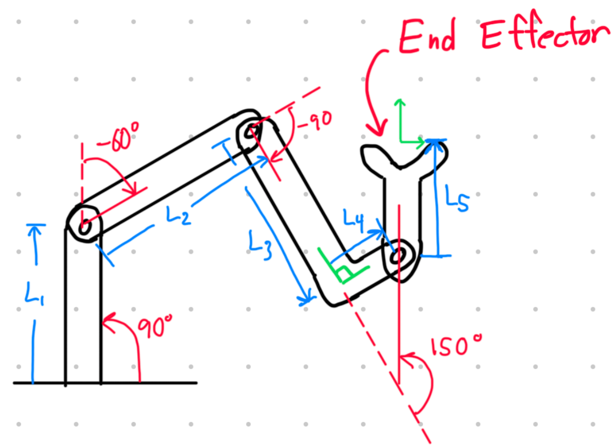
$$\text{rotation} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \quad , \qquad \text{translation} = \begin{bmatrix} t_{forward} \\ t_{left} \end{bmatrix}$$

When you chain n of these *homogeneous transformations* together (like $\mathbf{T}_{\text{final}} = \mathbf{T}_0\, \mathbf{T}_1 \ldots \mathbf{T}_{n\text{-}1}\, \mathbf{T}_n$) then the resulting matrix $\mathbf{T}_{\text{final}}$ encodes both the position and orientation of the very end!

2) Application

Let's see this applied to a robotic arm:

```
% Link lengths
L1= .1;
L2= .1;
L3= .1;
L4= 0.05;
L5 = 0.05;

% Anonymous active rotation function, using degrees
instead of radians
rotd = @(x) [cosd(x) -sind(x); sind(x) cosd(x)];

% Angles
theta_0 = 90;
theta_1 = -60;
theta_2 = -90;
theta_3 = 150;

% Relative homogenous transformation
T0 = [rotd(theta_0) [0; 0];   0 0 1];
T1 = [rotd(theta_1) [L1; 0];  0 0 1];
T2 = [rotd(theta_2) [L2; 0];  0 0 1];
T3 = [rotd(theta_3) [L3; L4]; 0 0 1];
T4 = [rotd(0)       [L5; 0];  0 0 1];

% Homogeneous transformation for location of End Effector
T_EE = T0*T1*T2*T3*T4

% Extract the end-effector orientation and position from the homogeneous transformation
ee_orientation = T_EE(1:2,1:2)
ee_position = T_EE(1:2,3)
```



Run the above code in Matlab, and look at how the output is the end effector's location. (Feel free to check this through a classic trigonometry approach.) Note how the output derives from the final matrix, `T_EE`. Specifically:

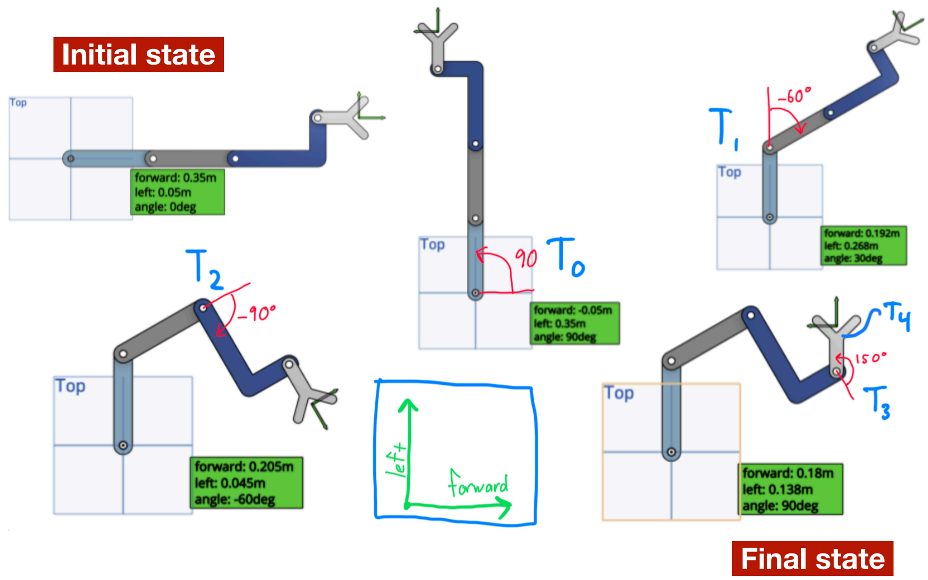
```
T_EE =

        0   -1.0000    0.1799
   1.0000         0    0.1384
        0         0    1.0000
```

And from this, you can see the 2x2 rotation matrix (look at a rotation matrix and think “What’s **cos(90°)**?” “What’s **sin(90°)**?”) and the 2D vector $[0.1799\ 0.1384]^T$.

3) Description

What is going on is each homogeneous transformation can be thought of as 1) moving a direction and then 2) turning to face a new direction. So:

- $\mathbf{T}_0$ moves 0 meters forward, 0 to the left, and then rotates +90 degrees.
- $\mathbf{T}_1$ moves L_1 meters forward, 0 to the left, and then rotates -60 degrees.
- $\mathbf{T}_2$ moves L_2 meters forward, 0 to the left, and then rotates -90 degrees.
- $\mathbf{T}_3$ moves L_3 meters forward, L_4 to the left, and then rotates +150 degrees.
- $\mathbf{T}_4$ moves L_5 meters forward, 0 to the left, and doesn’t rotate.



We multiply all those together left to right ($\mathbf{T}_{ee} = \mathbf{T}_0\ \mathbf{T}_1\ \mathbf{T}_2\ \mathbf{T}_3\ \mathbf{T}_4$), and the resulting matrix ($\mathbf{T}_{ee}$) has the end effector’s 2-D *pose* (i.e. orientation **and** position).

(We drew this in OnShape, check out the document at https://cad.onshape.com/documents/10699136a3cc197fc422327f/w/b35bb4b2c906434c20a78052/e/730f26bbf1e1881db93b2f00)

4) Homework question

Now, we’re going to think about a completely different object than a robotic arm, but the principles don’t change in the slightest. We’ll look at the Green Line map, and determine where the end is based on the various turns and straightaways.

Let’s assume that we have the following T stops, with the following relative distances and angles between them.

1. BU Central direction: **-45 degrees**
2. BU Central to Kenmore: **1km then +45 degrees**
3. Kenmore to Boylston: **2km then +45 degrees**
4. Boylston to Government Center: **1km**

a): Find the **location of Government Center relative to the origin** (Boston University Central). You’ll use the identical process as above for the arm, **only the angles and distances will change**.

- Please show your work by hand
- You must use the above *homogeneous transformation* approach
- You must show us the final matrix, from which you extract the location
- You can check your work with CAD, even with basic high-school trig. Nonetheless, you **must** use homogeneous transformations for the final answer.

Q4 Matlab

You know from Class 24 in EK121/EK122 that multiplying matrices in Matlab is straightforward, e.g. “A times x” is literally written `A*x`. In this next section, you have a list of transformation matrices which you must multiply times the original image to see the outcome.

For this question, you must:

1. use a **for loop** to iterate (covered in Class 26 in EK121)
2. use a **subplot** to draw the results (you used subplots in the final EK121 project)

Please submit the following:

A. Publish your code and the resulting **TWO** graphs. The graphs should look similar to the attached image (but without the red graph numbers, and you don’t need the circular markers to be red)
B. Using Example 3 and Tables 1-4 in Section 1.9, name each geometric transformation. For instance, if you believe plot #2 comes from a “rotation by 90 degrees”, then write as such. **Make sure you label your answers #1-8, corresponding to the numbered graphs below.**