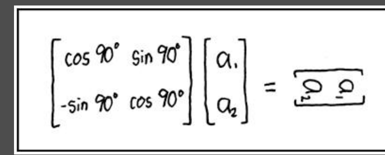
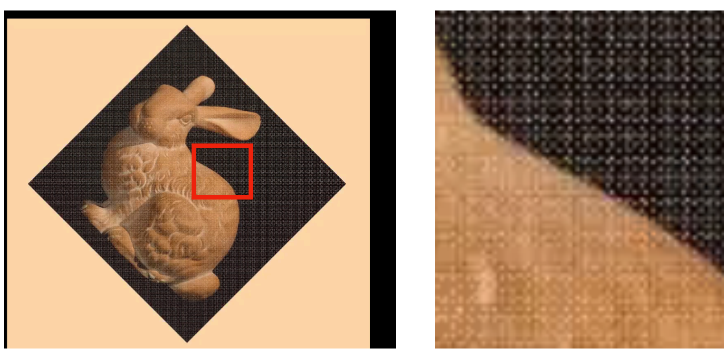
Transformations in 2D
Short version

Transformations

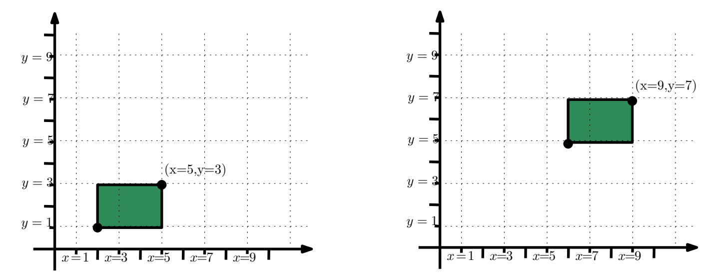


Something to be careful about with hw1



Translations (shift) by(α, β)

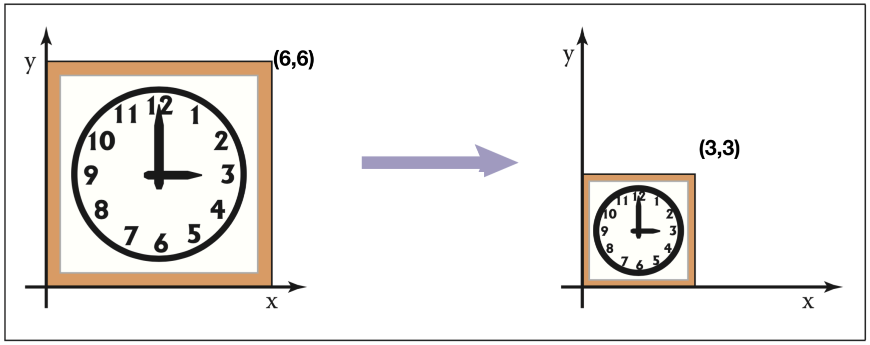
Translation (shift) by $(4, 4)$
$(x, y) \rightarrow (x + 4, y + 4)$



- Adding a constant α to the x-coordinate of every point
- Adding a constant β to the y-coordinate of every point
- $(x, y) \rightarrow (x + \alpha, y + \beta)$

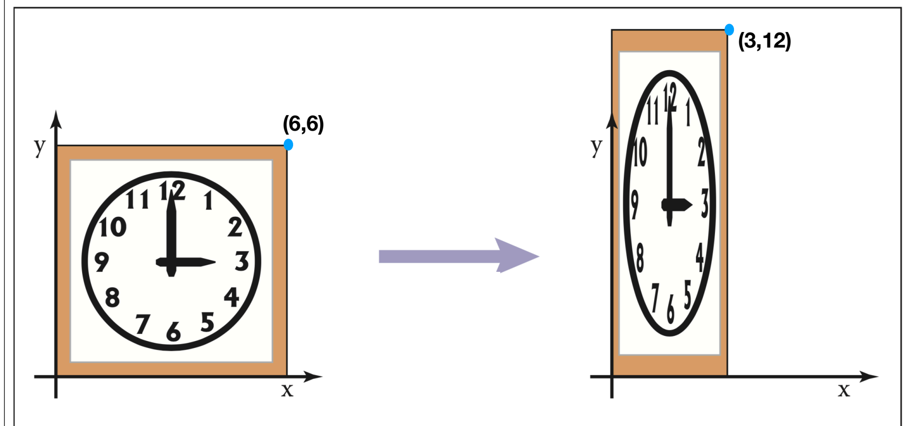
Scaling

- We can use two constants (s_x, s_y) for the x-axis and the y-axis. Then we shift each point (x,y) into the point $(s_x \cdot x,\ \ s_y \cdot y)$
- $(x,y) \rightarrow (s_x \cdot x,\ \ s_y \cdot y)$
- Example $(x,y) \rightarrow (x/2,\ y/2)$



Scaling

- Example: $(x,y) \rightarrow (0.5x,\ \ 2y)$



The mathematician and coffee cup non-funny joke Part 1

Fence

Solution:
1. **Walk around the fence,**
2. **fetch coffee kettle,**
3. **walk back pure coffee,**
4. **drink**

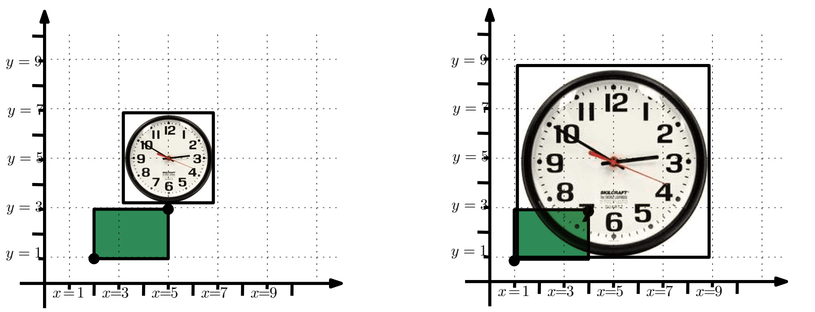
The mathematician and coffee cup non-funny joke Part 2

Fence

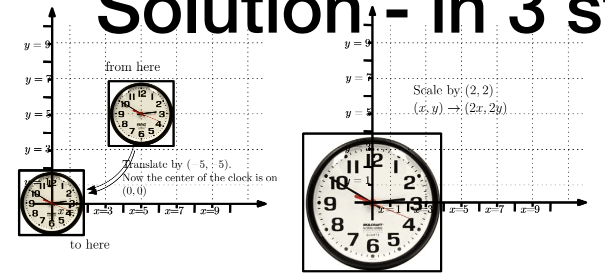
Solution:
1. **Bring the coffee Kettle to the other table, and walk to the left table**
2. **Apply the solution from the previous slide**

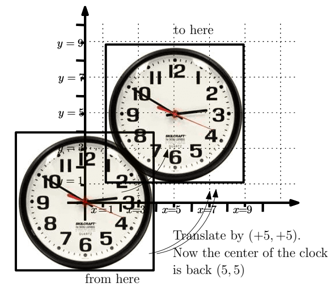
Resize the clock, without changing its center

Problem: scale the clock, but without changing its center and without effecting the green rectangle



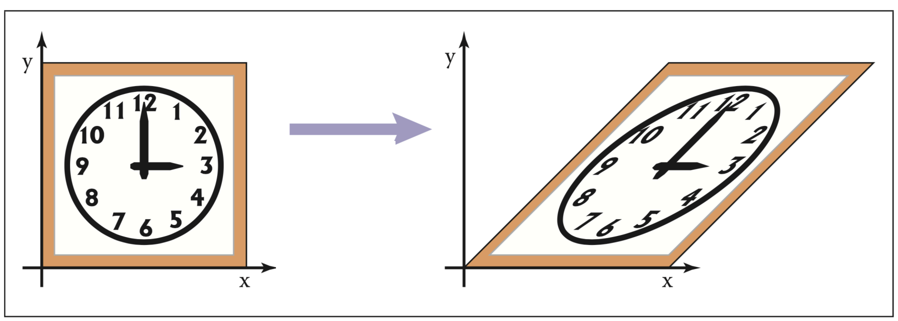
Solution - in 3 steps





Shearing

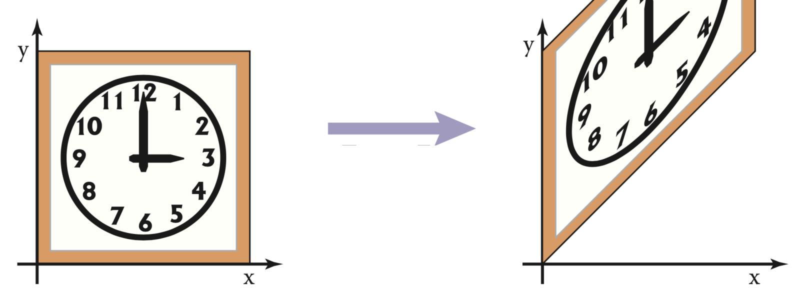
- If we move each point (x,y) into the point $(x, y) \to (x + y,\ y)$



Shearing

- Vertical shearing shifts each column based on the x value.

$$(x, y) \to (x,\ x + y)$$

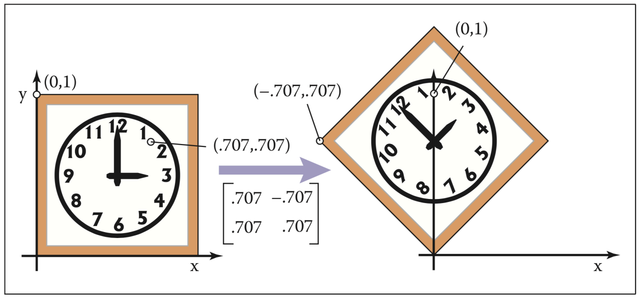


Rotation

- Rotate counterclockwise by an angle θ about the origin.

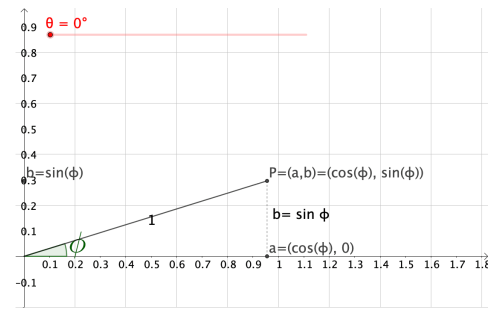
$(x, y) \rightarrow (x\cos\theta - y\sin\theta,\ \ x\sin\theta + y\cos\theta)$

New *x* **New *y***



Assume we rotate p by an angle θ CCW

Starting from a point $P = (a, b)$, where will this point find itself after rotation by θ in the CounterClockwise direction ?
Let $p' = (x', y')$ denote the new location of this point. Lets compute this location:
For simplicity, assume $a^2 + b^2 = 1$

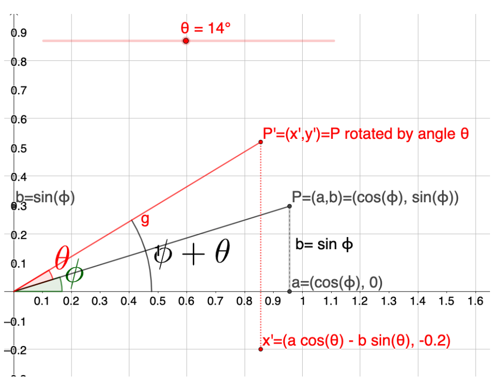


$x' = \cos(\phi + \theta) =$
$\underbrace{\cos(\phi)}_{=a}\cos(\theta) - \underbrace{\sin(\phi)}_{=b}\sin(\theta)$
$= a\cos(\theta) - b\sin(\theta)$

$y' = \sin(\phi + \theta) =$
$\underbrace{\sin(\phi)}_{=b}\cos(\theta) + \underbrace{\cos(\phi)}_{=a}\sin(\theta)$
$a\sin(\theta) + y\sin(\theta)$

Assume we rotate p by an angle θ CCW

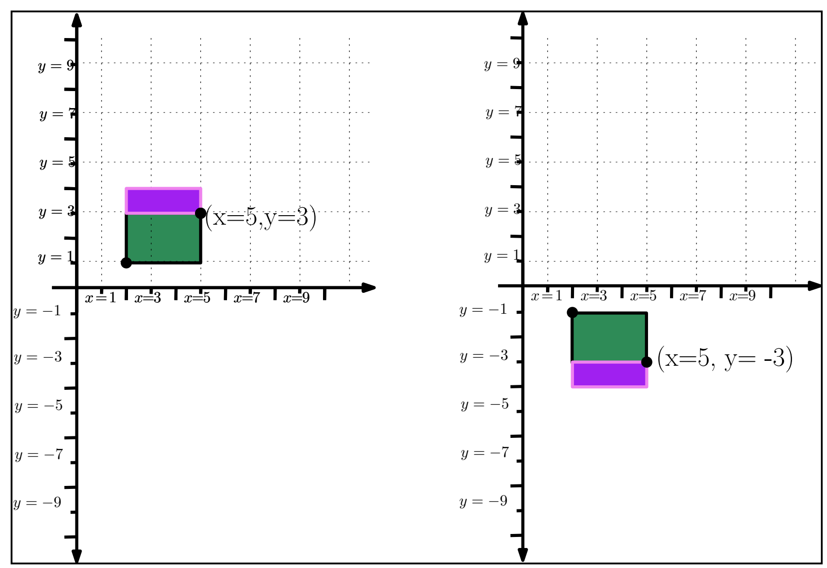
Starting from a point $P = (a, b)$, where will this point find itself after rotation by θ in the CounterClockwise direction ?
Let $p' = (x', y')$ denote the new location of this point. Lets compute this location:
For simplicity, assume $a^2 + b^2 = 1$



$x' = \cos(\phi + \theta) =$
$\underbrace{\cos(\phi)}_{=a}\cos(\theta) - \underbrace{\sin(\phi)}_{=b}\sin(\theta)$
$= a\cos(\theta) - b\sin(\theta)$

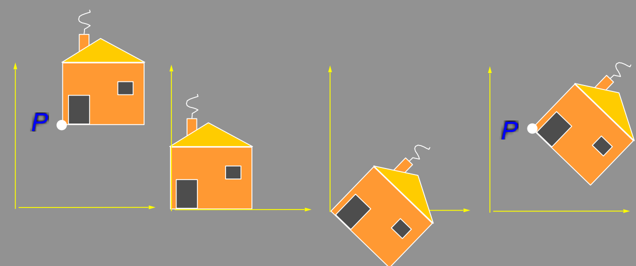
$y' = \sin(\phi + \theta) =$
$\underbrace{\sin(\phi)}_{=b}\cos(\theta) + \underbrace{\cos(\phi)}_{=a}\sin(\theta)$
$a\sin(\theta) + y\sin(\theta)$

Reflection on the x-axes: $(x, y) \rightarrow (x, \ -y)$



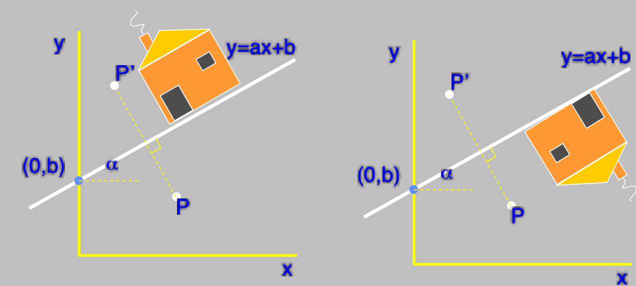
Transformation Composition

- What operation rotates by θ around $P = (p_x, p_y)$?
  - Translate P to origin
  - Rotate around origin by θ
  - Translate back



Arbitrary Reflection - promo
We will get back to it later in the semester

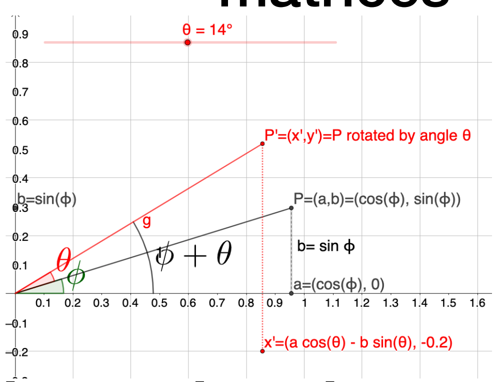
$\alpha = \tan^{-1}(a)$



1. Compute b.
2. Shift by (0,-b)
3. Rotate by $-\alpha$ CCW
4. Reflect through x
5. Rotate by α
6. Shift by (0,b)

Very scarrrrry….
Unless we represent transformation by matrices
And then it is trivial

Expressing rotations with matrices



- $$R_\theta = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \cdot \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} a\cos\theta - b\sin\theta \\ a\sin\theta + b\cos\theta \end{bmatrix} = \begin{bmatrix} a' \\ y' \end{bmatrix}$$

What about other operaions

- Scaling by α ? $(x, y) \to (\alpha x, \alpha y)$.
- $M = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix}$, and $p = \begin{pmatrix} x \\ y \end{pmatrix}$. Then
  $Mp = M \begin{pmatrix} x \\ y \end{pmatrix} = \begin{bmatrix} \alpha & 0 \\ 0 & \alpha \end{bmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \alpha x \\ \alpha y \end{pmatrix}$
- Reflection by the x-axis ? $(x, y) \to (x, -y)$. $M = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
- Sheering ? e.g. $(x, y) \to (x, 2x + y)$ $M = \begin{bmatrix} 1 & 0 \\ 2 & 1 \end{bmatrix}$
- Translation is problematic ?

Concatenation

- A very common scenario - need to apply the same transformation on many points in the scene. (the same transformation applied to each point).
- Recall - matrix multiplication is associative $A(B \cdot C) = (A \cdot B) \cdot C = A \cdot B \cdot C$?
- So for a point p, we can understand the expression $(M_3 \cdot M_2 \cdot M_1)p$ as a three step process
- Apply the transformation M_1 on p, (that is, compute $M_1 \cdot p$. Then
- Apply M_2 on the result. That is, compute $M_2 \cdot M_1 \cdot p$.
- Apply M_3 on the result - that is, computer $M_3 \cdot (M_2 \cdot (M_1 p))$
- Alternatively (and usually more efficient) - compute $M' = M_3 \cdot M_2 \cdot M_1$, and for every point p, compute $M' \cdot p$.

Homogeneous coordinates

- We represent a point $p = (x, y)$ using 3 numbers $p = (x, y, w)_h$
- What are the coordinates of this point in Euclidean Cartesian representation ? $p = (x/w,\ y/w) = (x, y, w)_h$
- So $(4,2)_{Cartesian} = (4,2,1)_{homog} = (8,\ 4,\ 2)_h = (2,\ 1,\ 0.5)_{homog}$
- Warning, this is a point in 2D, (not a point in $\mathbb{R}^3$). On the other hand…
- The point $(4,2,8)_{Cartesian} = (4,2,8,1)_{homog} = (8,\ 4,\ 16,\ 2)_h = (2,\ 1{,}4,\ 0.5)_{homog}$ is a point in 3D.

Homogenous transformations are extremely useful in multiple graphics settings - including translations

$$\begin{bmatrix} 1 & 0 & \alpha \\ 0 & 1 & \beta \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix}_h = \begin{bmatrix} x_0 + \alpha \cdot 1 \\ y_0 + \beta \cdot 1 \\ 1 \cdot 1 \end{bmatrix} =$$

- That is, this transformation performs translation by α and β: $(x, y) \to (x + \alpha, y + \beta)$

https://www.geogebra.org/classic/hpqxbcmd

Homogenous transformations cont - the matrices of the other transformation

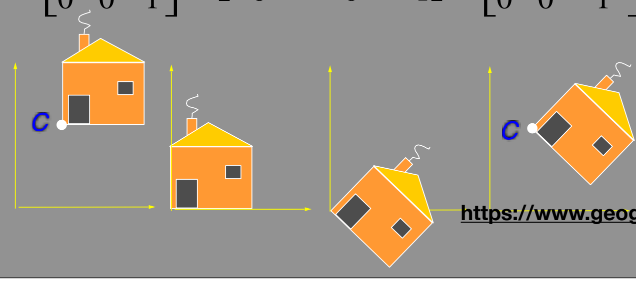
- If $M = \begin{bmatrix} a & b & 0 \\ c & d & 0 \\ 0 & 0 & 1 \end{bmatrix}$ then $M \cdot \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$ is just like $\overbrace{\begin{bmatrix} a & b \\ c & d \end{bmatrix}}^{M'} \begin{bmatrix} x \\ y \end{bmatrix}$. That is, we can ignore the red part
- Example $\overbrace{\begin{bmatrix} 1/2 & 0 \\ 0 & 1/2 \end{bmatrix}}^{M'} \begin{bmatrix} x \\ y \end{bmatrix}$ Scaling by 1/2. Have the same effect as $M = \begin{bmatrix} 1/2 & 0 & 0 \\ 0 & 1/2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$
- $M = \begin{bmatrix} a & b & \alpha \\ c & d & \beta \\ 0 & 0 & 1 \end{bmatrix}$ has the same effect as (first) apply $\overbrace{\begin{bmatrix} a & b \\ c & d \end{bmatrix}}^{M'}$ on p, and then translate by (α, β).
- Example: $M = \begin{bmatrix} 1/2 & 0 & 2 \\ 0 & 1/2 & 3 \\ 0 & 0 & 1 \end{bmatrix}$ scale by 1/2, and then translate by (2,3)
- In most cases, the last row of M is $[0,0,1]$. We will change it only when discussing projections.

gg

Transformation Composition

- What operation rotates by -θ around $C = (x_0, y_0)$
  - Translate by $-(x_0, y_0)$. It translate C to the origin
  - Rotate around origin by -θ
  - Translate back
  - $M = \left(Trans(C) \cdot Rotate(\theta) \cdot Trans(-C)\right)p$

$$M = \begin{bmatrix} 1 & 0 & x_0 \\ 0 & 1 & y_0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} cos\theta & -sin\theta & 0 \\ sin\theta & cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & -x_0 \\ 0 & 1 & -y_0 \\ 0 & 0 & 1 \end{bmatrix}$$



https://www.geogebra.org/m/mtev9

Identity Matrix and Inverse matrix

- The matrix $I = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ is called the identity matrix.
- Note that for every matrix M, it holds that $M \cdot I = I \cdot M = 1$
- For a matrix M, we denote by M^{-1} a matrix M such that $M \cdot M^{-1} = I$
- Question: What is the inverse of $\begin{bmatrix} 1 & 0 & \alpha \\ 0 & 1 & \beta \\ 0 & 0 & 1 \end{bmatrix}$??

Rotations - more perspective (not in syllabus)

- If z_1, z_2 are complex numbers $z_1 = x_1 + iy_1,\ z_2 = x_2 + iy_2$
- Then $z_1 \cdot z_2$ is a new complex number, whose length is $|z_1 \cdot z_2| = |z_2| \cdot |z_2|$, and whose angle is the sum of angles of z_1, z_2

  $\arg(z_1 \cdot z_2) = \arg(z_1) + \arg(z_2)$

  We also know that $z_1 \cdot z_2 = x_1x_2 - y_1y_2 + i(x_1y_2 + x_2y_1)$

- Now, if $z_2 = \cos\theta + i\sin\theta$, (fixed for the transformation) and Z_1 is a pixel, then multiply z_1 by z_2 will not change the length of z_1 but it will change its argument. To be precise, it will rotate z_1 by $arg(z_2)$.

- then $z_1 \cdot z_2 = \underbrace{x_1\cos\theta - y_1\sin\theta}_{real\ part=x'} + i\underbrace{(x_1\sin\theta + y_1\cos\theta)}_{=y'}$

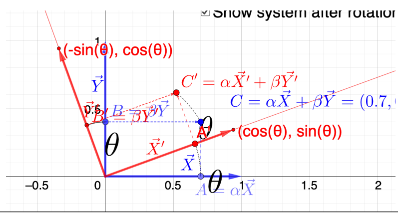
- This is useful when studying **quaternions** - (which are useful for 3D animation) youtube

Rotations - more perspectives
Transforming from one coordinate system to another

- From Linear algebra: A **basis** $\{\vec{v}_1, \vec{v}_2 \ldots \vec{v}_d\}$ is a set of vectors such that every point p in a space(plane/space…) could be expressed as a linear combination. $p = \alpha_1 \cdot \vec{v}_1 + \alpha_2\vec{v}_2 + \ldots + \alpha_d \cdot \vec{v}_d$ ….
- and in addition, we could not drop any of these vectors.
- The space is **spanned** by this basis.
- Multiplication by a matrix M is a linear operation: That is
  - $M \cdot \vec{0} = \vec{0}$
  - $M \cdot (\vec{u} + \vec{v}) = M\vec{u} + M\vec{v}$
  - $M(\alpha\vec{u}) = \alpha(M\vec{u})$
  - We are all very familiar with the basis $\overrightarrow{X} = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$ *and* $\overrightarrow{Y} = \begin{pmatrix} 0 \\ 1 \end{pmatrix}$

We can express rotation by creating a new basis of $\mathbb{R}^2$

- **To specify a rotation, it is sufficient to create a new coordinates system, and specify what is the correspondence between the old and new basis.**
- **To be precise, create M, such that the it's column of M is the $i'th$ vector (represented as a linear combination of the basis)**
- **The text above is probably very cryptic without multiple examples**
- **"Tricky" way to find rotation matrix. If $\overrightarrow{X}, \overrightarrow{Y}$ are unit vectors in the old coordinate system, then we could think abou rotation as rotating the coordinates systems as well, and after the rotation we expect**
- $\overrightarrow{X} \xrightarrow[by\ \theta]{Rotation} \overrightarrow{X'}$ **and** $\overrightarrow{Y} \xrightarrow[by\ \theta]{Rotation} \overrightarrow{Y'}$ **. Let R_θ be the rotation matrix. This means** $R_\theta \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \overrightarrow{X'}$ **and** $R_\theta \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \overrightarrow{Y'}$.
- **But** $R_\theta \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \overrightarrow{X'}$ **is just the first column of R_θ . And $R_\theta \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ is the second column.**
- **Lets try: Write** $R_\theta = \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix}$.
- **then for every data point $C = (\alpha, \beta)$, we could write (in a somehow obnoxious way)** $C = \alpha \overrightarrow{X} + \beta \overrightarrow{Y}$.
- $R_\theta \cdot C = \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \left(\alpha \overrightarrow{X} + \beta \overrightarrow{Y} \right) \overset{\text{linearity}}{=} \alpha \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \overrightarrow{X} + \beta \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \overrightarrow{Y} = \alpha \overrightarrow{X'} + \beta \overrightarrow{C'}$



We can express rotation by creating a new basis of $\mathbb{R}^2$

- This is going to be extremely useful when discussing rotations in 3D
- To specify a rotation, it is sufficient to create a new coordinates system, and specify what is the correspondence between the old and new basis.
- To be precise, create M, such that the it's **column** of M is the $i'th$ vector (represented as a linear combination of the old basis)
- The text above is probably very cryptic without multiple examples
- "Tricky" way to find rotation matrix. If $\overrightarrow{X}, \overrightarrow{Y}$ are unit vectors in the old coordinate system, then we could think about the rotation as rotating the coordinates systems as well, and after the rotation we expect
- $\overrightarrow{X} \xrightarrow[by\ \theta]{Rotation} \overrightarrow{X'}$ and $\overrightarrow{Y} \xrightarrow[by\ \theta]{Rotation} \overrightarrow{Y'}$. Let R_θ be the rotation matrix. This means $R_\theta \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \overrightarrow{X'}$ and $R_\theta \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \overrightarrow{Y'}$.
- But $R_\theta \cdot \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \overrightarrow{X'}$ is just the first column of R_θ . And $R_\theta \cdot \begin{bmatrix} 0 \\ 1 \end{bmatrix}$ is the second column.
- Lets try: Write $R_\theta = \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix}$.
- then for every data point $C = (\alpha, \beta)$, we could write (in a somehow obnoxious way) $C = \alpha \overrightarrow{X} + \beta \overrightarrow{Y}$.
- $R_\theta \cdot C = \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \left(\alpha \overrightarrow{X} + \beta \overrightarrow{Y} \right) \overset{\text{linearity}}{=} \alpha \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \overrightarrow{X} + \beta \begin{bmatrix} \vdots & \vdots \\ \overrightarrow{X'} & \overrightarrow{Y'} \\ \vdots & \vdots \end{bmatrix} \overrightarrow{Y} = \alpha \overrightarrow{X'} + \beta \overrightarrow{C'} = C'$
- Important take home message: To find the rotation matrix, just create a matrix where each column is one of the new basis vector (written using coordinates of the old coordinate system)