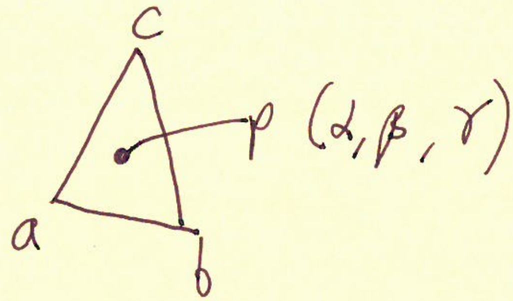
Jan 30



$p.red = \alpha * a.red + \beta * b.red + \gamma * c.red$

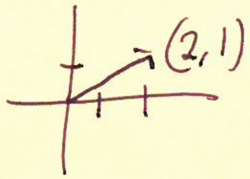
$p.green = \alpha * a.green + \beta * b.green + \gamma * c.green$

$p.blue = \ldots$

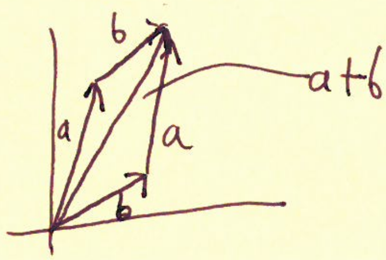
Some L.A.

Geometric idea of a vector

direction & magnitude



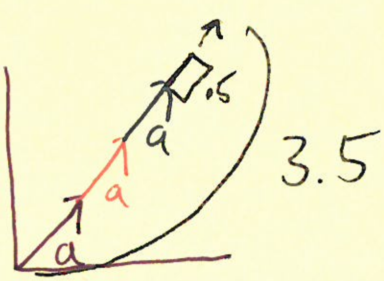
vector addition: $\begin{bmatrix} a_x \\ a_y \end{bmatrix} + \begin{bmatrix} b_x \\ b_y \end{bmatrix} = \begin{bmatrix} a_x + b_x \\ a_y + b_y \end{bmatrix}$



we show

$a + b = b + a$

scaling: $\alpha \begin{bmatrix} a_x \\ a_y \end{bmatrix} = \begin{bmatrix} \alpha a_x \\ \alpha a_y \end{bmatrix}$

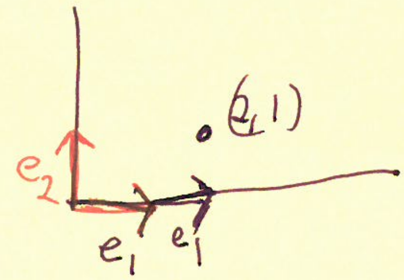


Cartesian Coordinates

$$e_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad e_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad \text{in general} \quad e_i = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

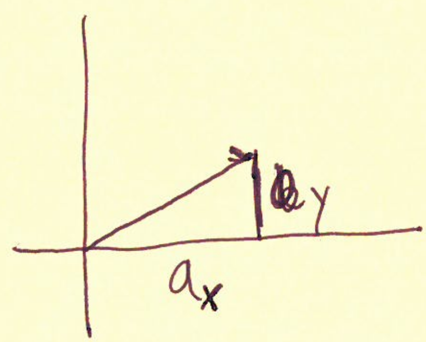
1 in i^{th} pace

$$a = \begin{bmatrix} a_x \\ a_y \end{bmatrix} = a_x e_1 + a_y e_2$$



Vector length

$$\|a\| = \sqrt{a_x^2 + a_y^2}$$



Vector transpose

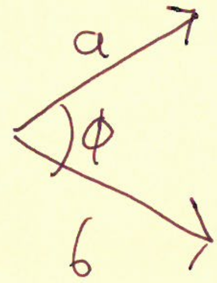
$$a = \begin{bmatrix} a_x \\ a_y \end{bmatrix} \qquad a^T = [a_x \ a_y] \qquad \text{in general}$$

$$a_{ij} \rightarrow a_{ji}$$

Dot products $a, b \in \mathbb{R}^2$

$$\cancel{a \cdot b = \sum_i a_i b_i}$$

$$a \cdot b = \|a\| \, \|b\| \cos \phi$$



props:

$a \cdot b = b \cdot a$ (commutative)

$a \cdot (b + c) = (a \cdot b) + (a \cdot c)$ (distributive over vector addition)

$\alpha \in \mathbb{R}$ $\cancel{a}$ (scalar mult)

$$(\alpha a) \cdot b = a \cdot (\alpha b) = \cancel{\alpha a} = \alpha (a \cdot b)$$

How do we get from

$$a \cdot b = \|a\| \|b\| \cos \phi$$

to $a \cdot b = \sum_i a_i * b_i$?

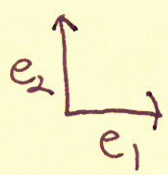
$a = a_x e_1 + a_y e_2 \qquad b = b_x e_1 + b_y e_2$

$$a \cdot b = (a_x e_1 + a_y e_2) \cdot (b_x e_1 + b_y e_2)$$

$$= a_x b_x (e_1 \cdot e_1) + a_x b_y (e_1 \cdot e_2)$$

$$a_y b_x (e_2 \cdot e_1) + a_y b_y (e_2 \cdot e_2)$$

$$= a_x b_x + a_y b_y$$



$$e_1 \cdot e_1 = 1 * 1 * \cos(0) = 1 * 1 * 1 = 1$$

$$e_2 \cdot e_2 = 1$$

$$e_1 \cdot e_2 = 1 * 1 * \cos\left(\frac{\pi}{2}\right) = 1 * 1 * 0 = 0$$

$$e_2 \cdot e_1 = 0$$

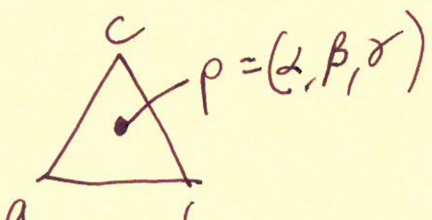
Matrix vector products

$$Ax = y$$

$$m\times n \quad \underset{?}{\boxed{n}}\times 1 = \underset{?}{\boxed{m}} \times \underset{?}{\boxed{1}}$$

$$\begin{bmatrix} a_{11} & a_{12} \\ a_{21} & a_{22} \\ a_{31} & a_{32} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} a_{11}\cdot x_1 + a_{12}\cdot x_2 \\ a_{21}\cdot x_1 + a_{22}\cdot x_2 \\ a_{31}\cdot x_1 + a_{32}\cdot x_2 \end{bmatrix}$$

Back to Barycentric coords



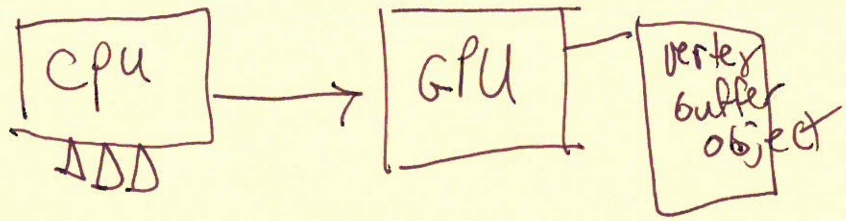
p.red = α a.red + β b.red + γ c.red

p.green = α a.green + β b.green + γ c.green

p.blue = α a.blue + β b.blue + γ c.blue

$$p.\text{color} = \begin{bmatrix} a.\text{red} & b.\text{red} & c.\text{red} \\ a.\text{green} & b.\text{green} & c.\text{green} \\ a.\text{blue} & b.\text{blue} & c.\text{blue} \end{bmatrix} \begin{bmatrix} \alpha \\ \beta \\ \gamma \end{bmatrix}$$

Vertex Buffer Objects



make a vetex buffer object w/ glGenBuffers

glGenBuffers(int n, GLuint *buffers)

gereate 2 vetex buffere

```
GLuint VBO[2];
glGenBuffers(2, VBO);
```

We want to tell openGL which buffer is the array buffer

glBindBuffer(GLenum target, GLuint buffer)

Ex

```
glBindBuffer(GL_ARRAY_BUFFER,
             VBO[1]);
```