

Vector lib
Friday, August 20, 2010
9:01 AM

Vectors is a vector lib ...
had to use whiteboard as I
couldn't get out of "select & Typo"
mode.

GO of lecture was:

- interface (.h files)
- implementation (.c file)

→ goal: build a .a lib that
your code can link against
just like it does when
linking with math lib:

linking with "main" ...
#include <math.h>
↑
interface
in /usr/include

cc main.c -lm

linking with
/usr/lib/libm.a

- we want to create a

vector.h & vector.c

in the same spirit so that
we can say cc main.c -lvec -lm

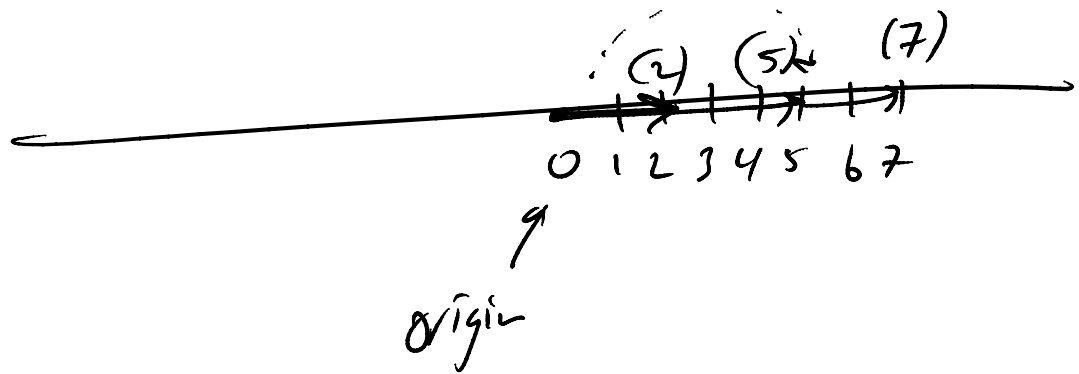
Vectors

Friday, August 20, 2010
10:23 AM

What are vectors? just representation
of numbers e.g. consider

$$\text{addition: } 5 + 2 = 7$$

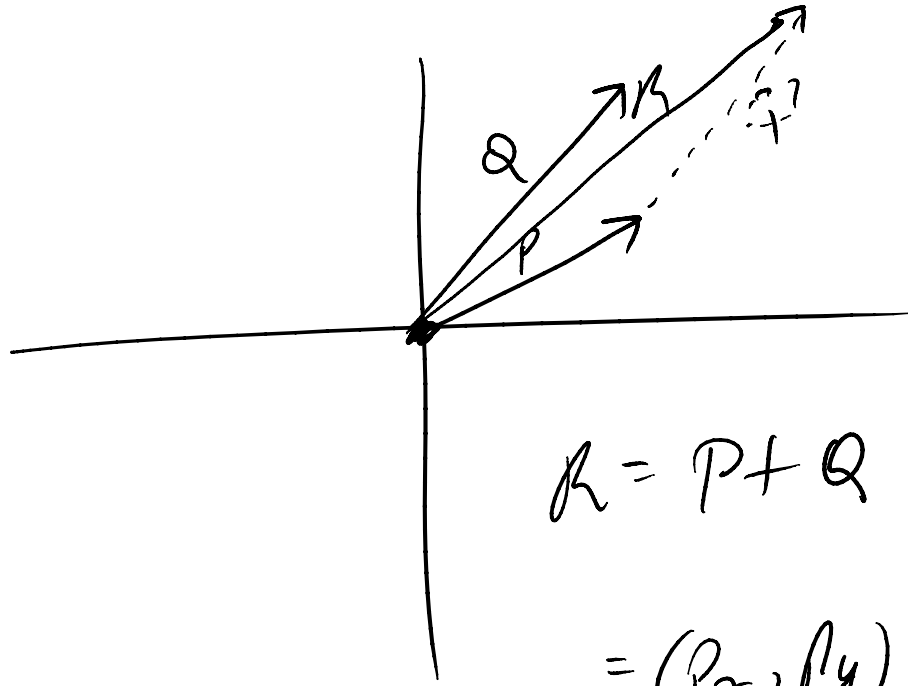
these are 1D vectors



we have

in 2D same thing, but per-component

57



$$R = P + Q$$

$$= (p_x, p_y) + (q_x, q_y)$$

$$= (p_x + q_x, p_y + q_y)$$

$$= (r_x, r_y)$$

our vector lib

Friday, August 20, 2010
10:26 AM

we need a type:

typedef double vec_t[3];

vec_t p; is the same as

double[3] p;

$\neq p \equiv (p_x, p_y, p_z) = \underbrace{(p[0], p[1], p[2])}$

away as
vector

So, $\text{vec_sum}(\text{vec_t } a, \text{vec_t } b, \text{vec_t } c)$
{

for ($i=0$; $i < 3$; $i++$)

- $c[i] = a[i] + b[i]$

$$e[i] = a[i] + b[i];$$

}