Learn You a Physics for Great Good!

>>> WORK IN PROGRESS <<<

Examples / Teeter

[src: Examples/Teeter.lhs] Previous: Single particle mechanics Table of contents Next: Box on an incline

```
module Examples.Teeter where
```

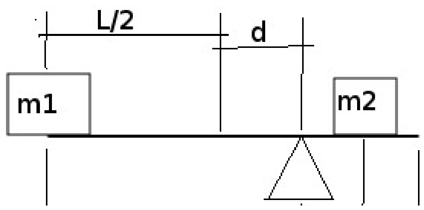
Exam excercise 3, 2017-01-13

```
import Dimensions.TypeLevel
import Dimensions.Quantity
import Prelude hiding (length)
```

Two boxes, m_1 and m_2, rests on a beam in balance.

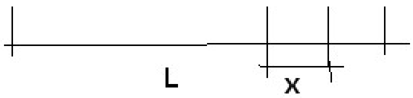
Known values:

```
beam_M = 1.0 # mass
m1 = 2.0 # mass
m2 = 5.0 # mass
d = 0.75 # length
beam_L = 5.0 # length
two = 2.0 # one
g = 9.0 # acceleration
```



Direct implication:

```
beam_left_L = (beam_L /# two) +# d
```




```
beam_right_L = beam_L -# beam_left_L
```

We want to be able to represenet the torques.

A torque (sv. vridmoment) is defined as:

$$\tau = distance\ from\ turning\ point \cdot force$$

$$\tau = distance\ from\ turning\ point \cdot mass \cdot gravitation$$

```
m1_torq = m1 *# (g *# beam_left_L)
```

To get the beams torque on one side, we need to divide by 2 because the beam's torque is spread out linearly (the density of the beam is equal everywhere), which means the left parts mass centre is $half\ the\ distance$ of the left parts total length.

$$beamL_\tau = beamL_M \cdot gravity \cdot \frac{beam\ left\ length}{2}$$

where

$$beamL_M = \frac{beam\ left\ length}{beam\ length} \cdot beam_M$$

```
beamL_torq = ((beam_left_L /# beam_L) *# (beam_M *# g)) *#
  (beam_left_L /# two)
```

```
beamR_torq = ((beam_right_L /# beam_L) *# (beam_M *# g)) *#
  (beam_right_L /# two)
```

We make an expression for $m2_\tau$, which involves our unknown distance x.

$$m2_\tau = m2 \cdot gravitation \cdot x$$

For the teeter to be in balance, both sides torques should be equal.

$$Left\ side\ torque = Right\ side\ angular\ torque$$

We try to break out $m2_\tau$ and then x.

$$m1_\tau + beamL_\tau = m2_\tau + beamR_\tau$$

$$m1_\tau + beamL_\tau - beamR_\tau = m2_\tau$$

$$the\ distance\ x = \frac{m2_{\tau}}{m2 \cdot gravitation}$$

Our solution:

```
x = (m1_torq +# beamL_torq -# beamR_torq) /# (m2 *# g)
```

Security check:

```
m2_torq = (m2 *# g) *# x
```

```
left_side_torque = m1_torq +# beamL_torq
right_side_torque = m2_torq +# beamR_torq
```

We can control that both sides total torque are equal, and that the dimensions of x is a length.

[src: Examples/Teeter.lhs] Previous: Single particle mechanics Table of contents Next: Box on an incline