```elm
import Keyboard
import Window

-- MODEL
areaW = 407
areaH = 301

hero : { x:Float, y:Float, vx:Float, vy:Float, dir:S
hero = { x=0, y=0, vx=0, vy=0, dir="south" }


-- UPDATE
velStep d obj =
    let f n = if d.x == 0 || d.y == 0 then toFloat n
    in  { obj | vx <- f d.x, vy <- f d.y }

dirStep {x,y} obj =
    { obj | dir <- if | x > 0 -> "east"
                      | x < 0 -> "west"
                      | y < 0 -> "south"
                      | y > 0 -> "north"
                      | otherwise -> obj.dir }

runStep running obj =
    let scale = if running then 2 else 1
    in  { obj | vx <- obj.vx * scale, vy <- obj.vy *

timeStep t ({x,y,vx,vy} as obj) =
    { obj | x <- clamp (-areaW/2) (areaW/2) (x + t *
            y <- clamp (-areaH/2) (areaH/2) (y + t *

step (time,arrows,run) hero =
    timeStep time . dirStep arrows . runStep run . v
```

Arrows to move
Shift to run

Hints: ☑   Options: ☐   Auto-update: ☑   [Hot Swap]   [Compile]