

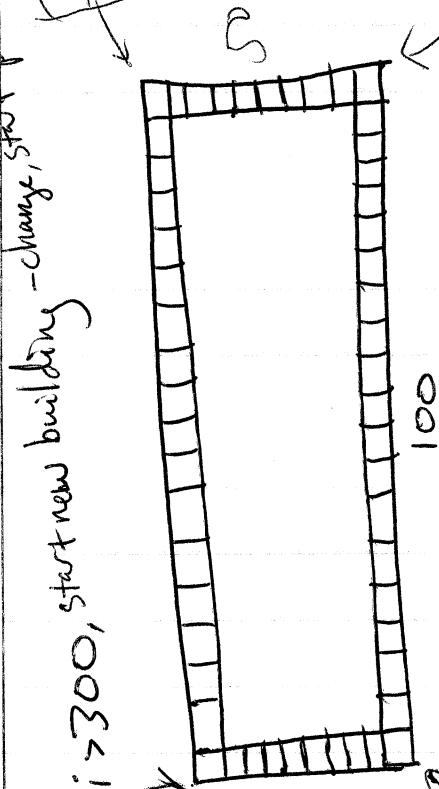
wallAmount is "i", could be very large #

wallLength $X = 100$

wallLength $Y = 50$

reset offsets ($x+0, y+4$)

reset offsets - change, start positions



default, $i=0, i++$

starts with offsets

$X+0$
 $Y+4$

$X+0$
 $Y+4$

- when $i > 50$
- change offsets

$X+4$
 $Y+0$

$i > \text{wallLength} Y$

- when $i > 150$
- change offsets

~~$X+0$~~
 ~~$Y-4$~~
 $X+0$
 $Y-4$

$i > \text{wallLength} Y \text{ and } X$

~~$Y+X$~~

- when $i > 200$
- change offsets

$X-4$
 $Y+0$

$i > \text{wallLength} X, Y, Y$
 ~~$Y+X$~~

multiple wallLengths
for different builds

wallLength $X1, Y1$
 $X2, Y2$

wallAmount = $(X1 + X2 + Y1 + Y2) \times 2$

Hit Left Side

$\text{heroRight} < \text{wallLeft}$
* $\text{heroBottom} > \text{wallTop}$

* $\text{heroTop} < \text{wallBottom}$
 ~~$\text{heroRight} < \text{wallLeft}$~~

Hit Right Side

* $\text{heroTop} < \text{wallBottom}$
 $\text{heroLeft} > \text{wallRight}$

* $\text{heroBottom} > \text{wallTop}$
 ~~$\text{heroLeft} > \text{wallRight}$~~

Hit Top Side

* $\text{heroLeft} < \text{wallRight}$
 $\text{heroBottom} < \text{wallTop}$

* $\text{heroRight} > \text{wallLeft}$

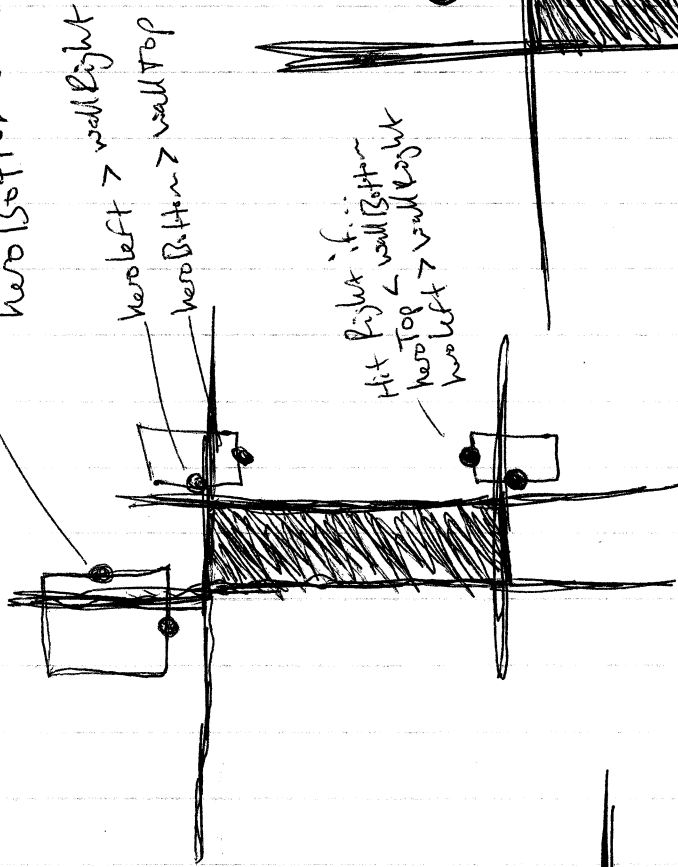
Hit Bottom Side

* $\text{heroLeft} < \text{wallRight}$
 $\text{heroTop} > \text{wallBottom}$

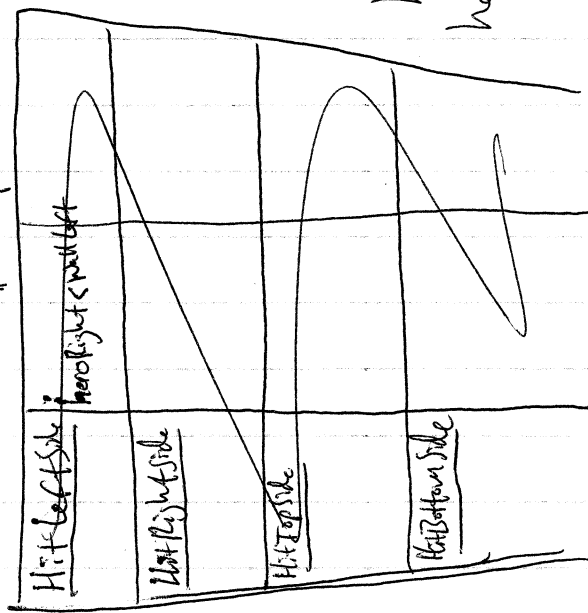
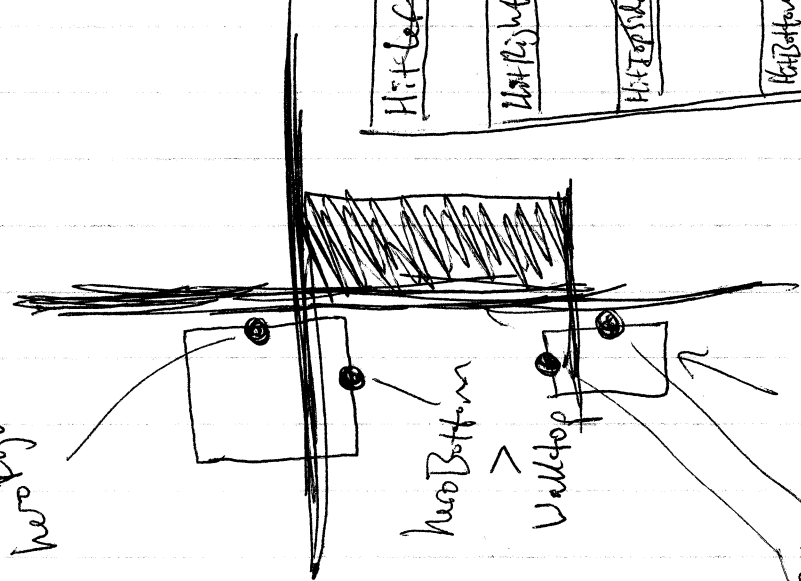
* $\text{heroRight} > \text{wallLeft}$

heroRight > wallLeft

heroBottom < wallTop

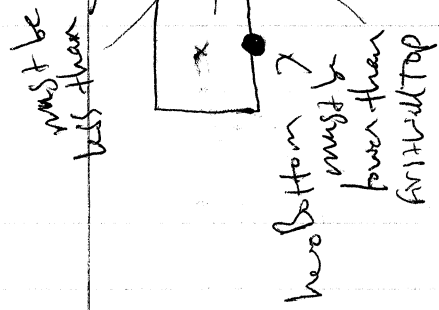
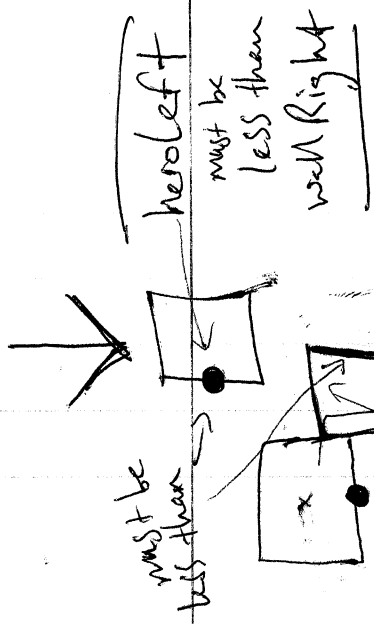


heroRight < wallLeft



Hit Bottom if
 $heroLeft < wallRight$
 $heroTop > wallBottom$

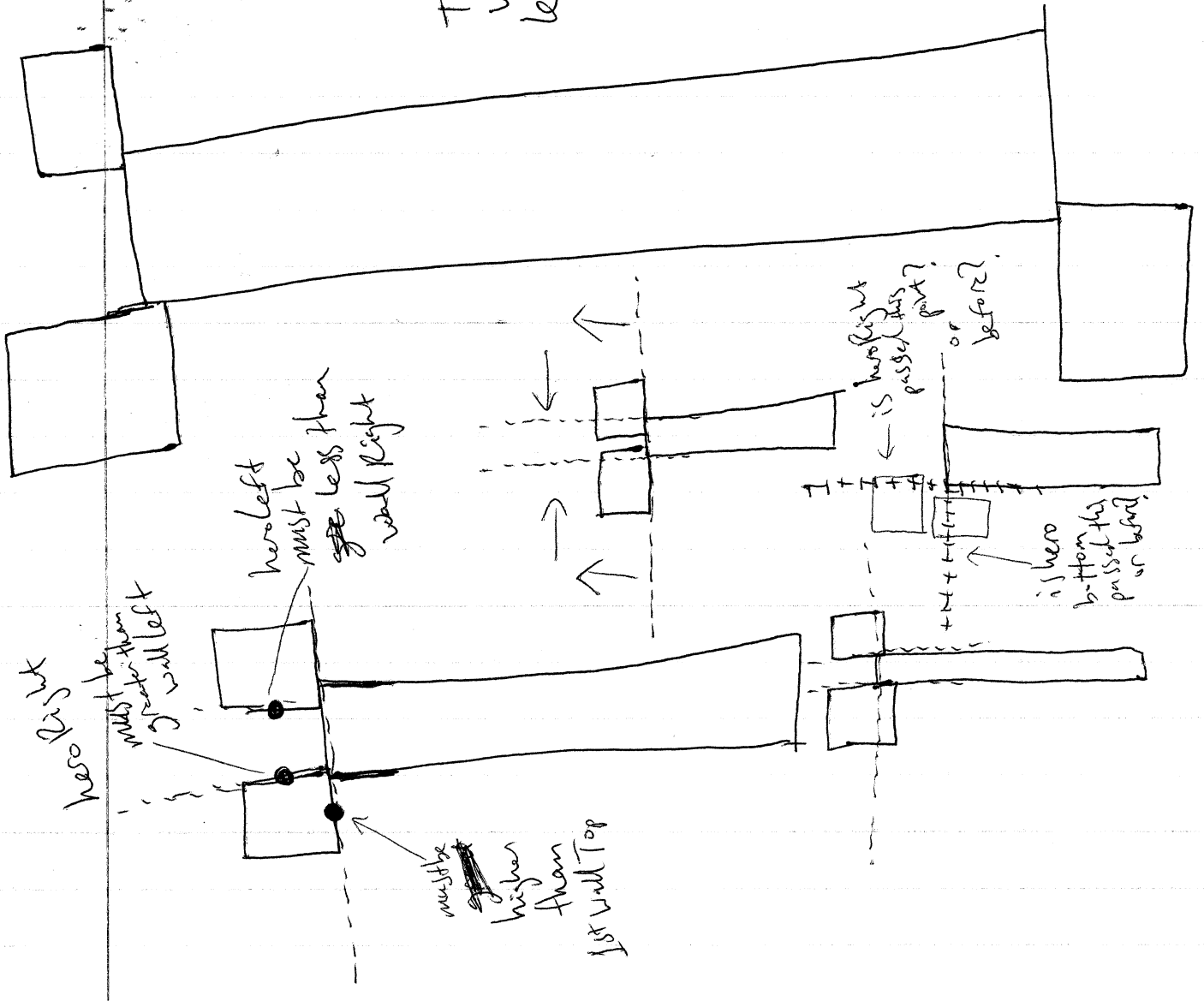
heroRight > wallLeft
heroTop > wallBottom



To collide with left side



heroTop < lastWallBottom
wallAt(ox, last.y)



Make wall/build

- current vars are used

- once ^{current} build is done, change current vars to next build

Use same function (copy) for horizontal walls and vertical walls.

So no X,Y offset, only X or Y ^{offset}

For Walls/builds

Position start (X,Y)

lengths of walls (X,Y)

Random Walls

~~next length~~ x 2

Already Calculated

- Start Bottom, Right, Top

- Wall Amount \leftarrow lengths added (x,y)

- Wall Amount \leftarrow lengths (x,y) x 2

focus on...

- hero

- walls

- doors

- windows

- bullets

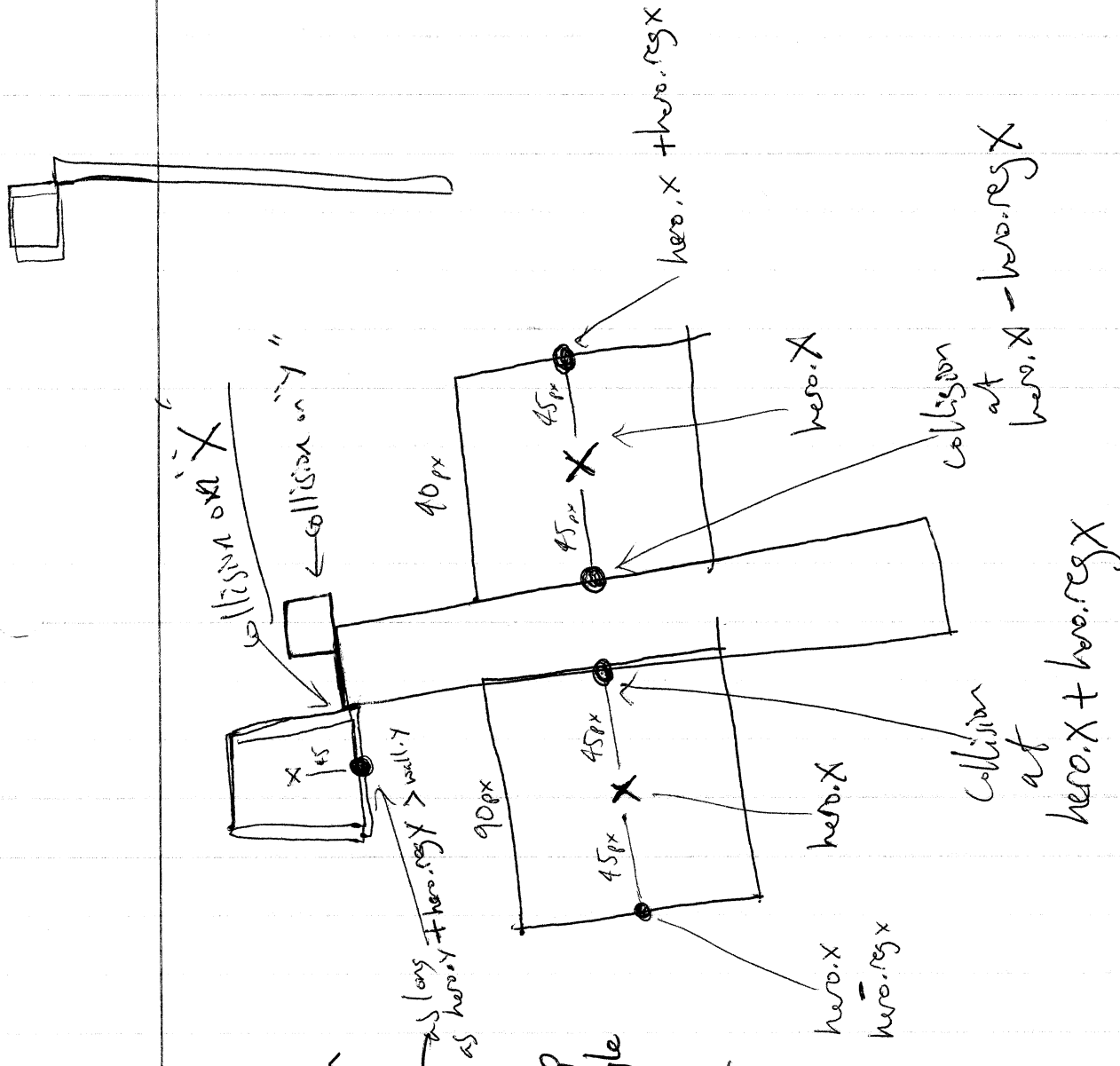
for now

option 1

use first and last
of array for collision
with end points

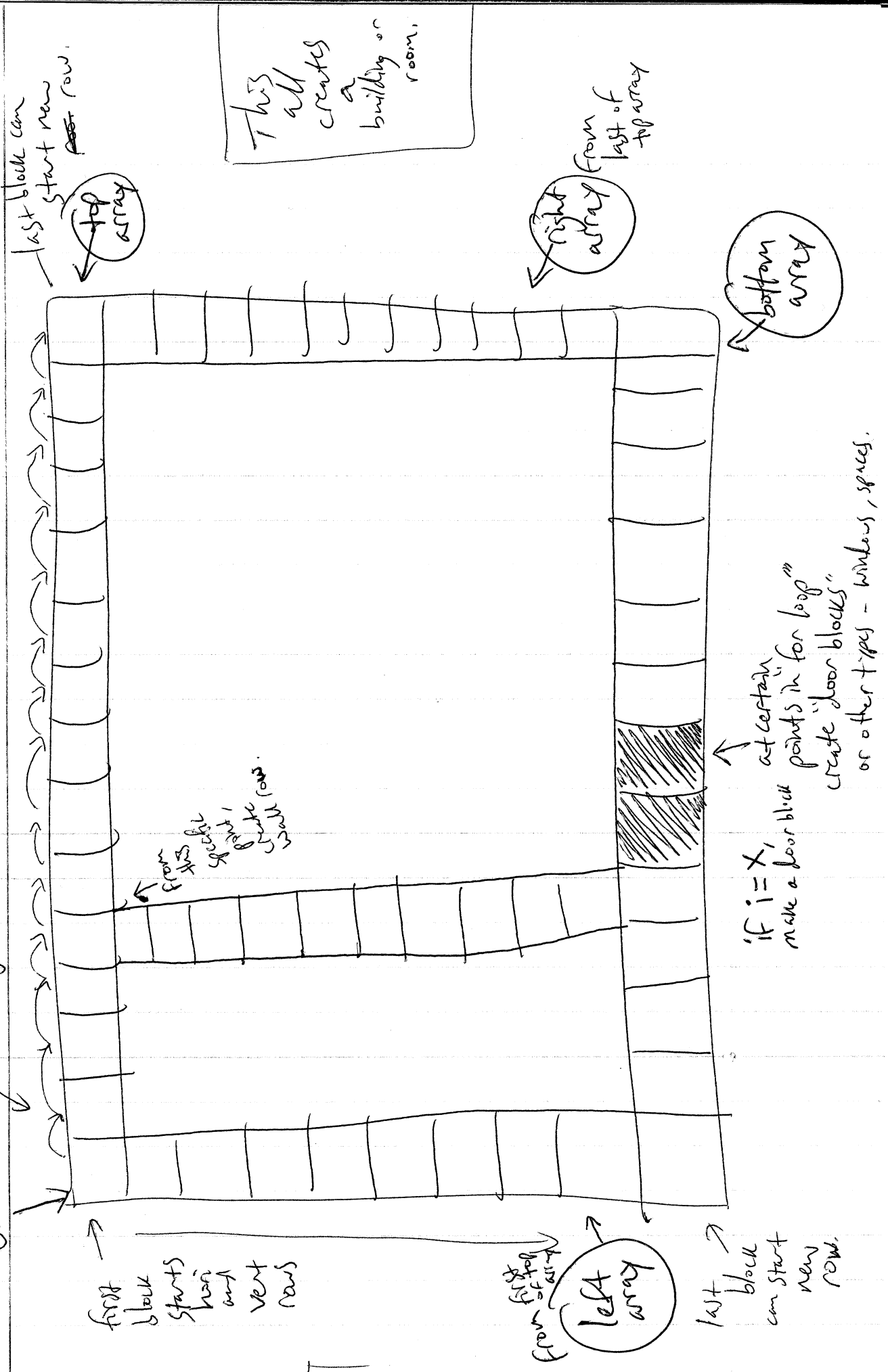
option 2

Somehow store or group
the array into a single
collidable object.

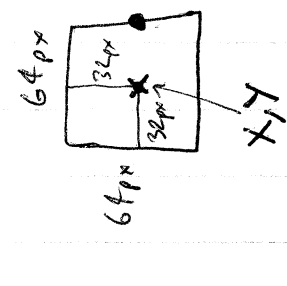
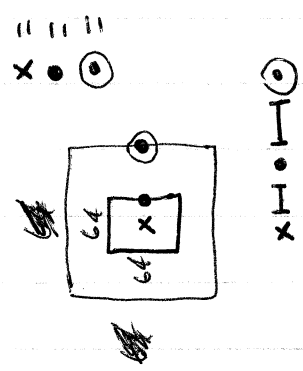


each block has a res point and hit box.

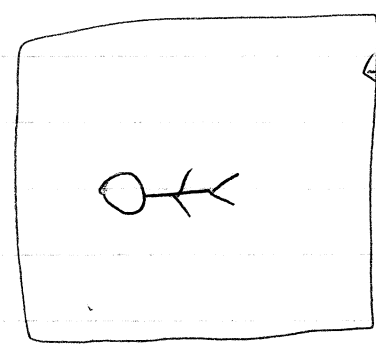
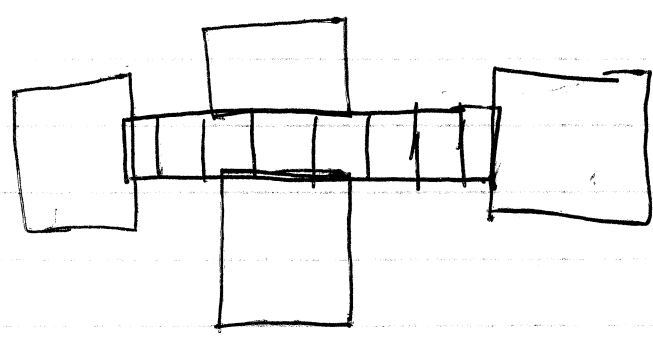
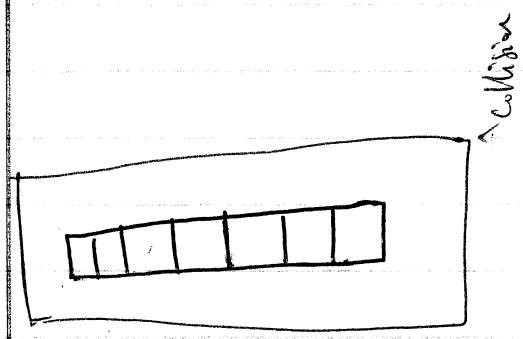
one block for loop
more blocks



if hero.x > 320



hero.x
hero.y
hero.reg.x = 32
hero.reg.y = 32

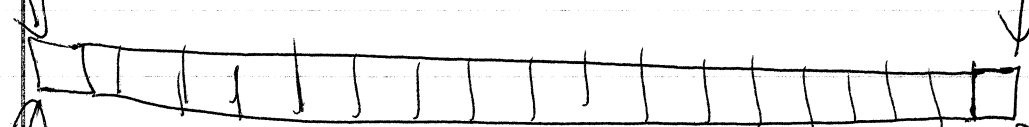


enemy sight
and
hearing
area

1st

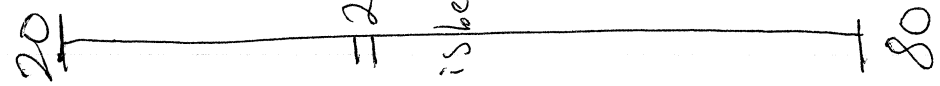
get
X and Y

X: 20
Y: 30



X: 20
Y: 80

last
get X and Y



if hero.X = 20 or in between for pulling
if hero.Y is between 20 and 80

Hits Top
 $heroBottom < wallTop$
 $heroRight > wallLeft$
 $heroLeft < wallRight$

Hits Left

Hits Right

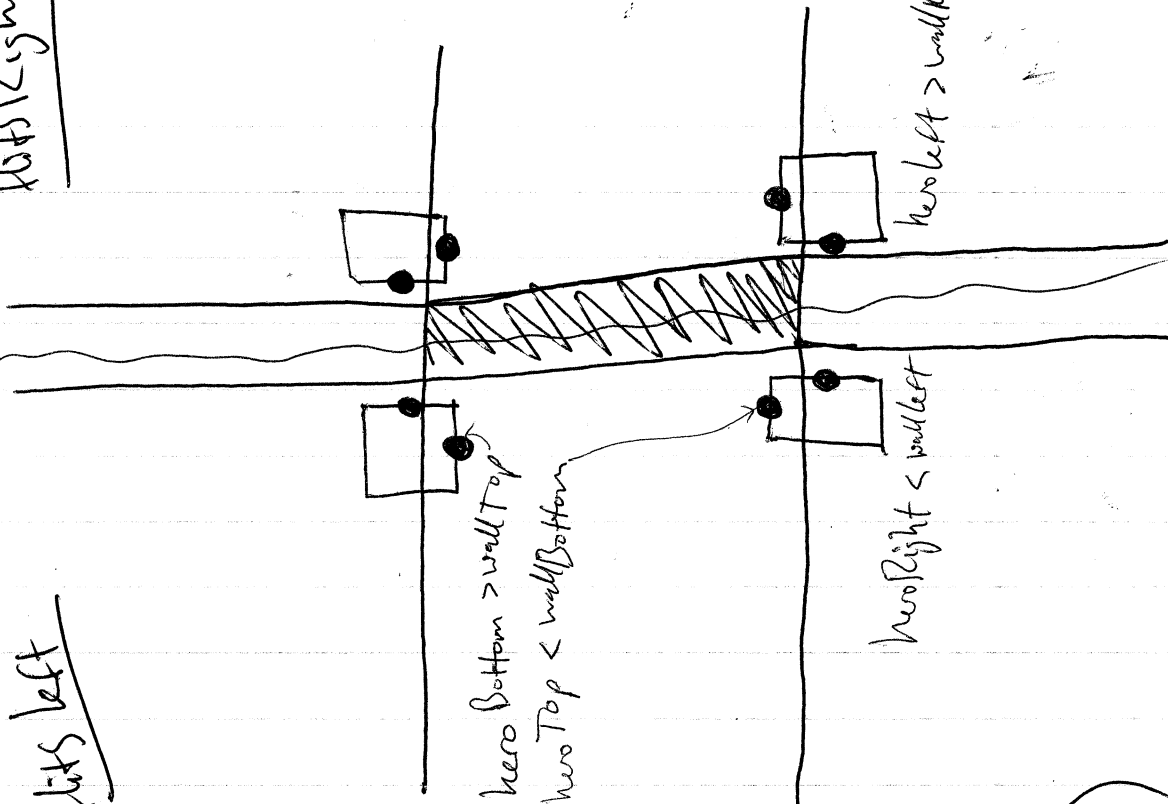
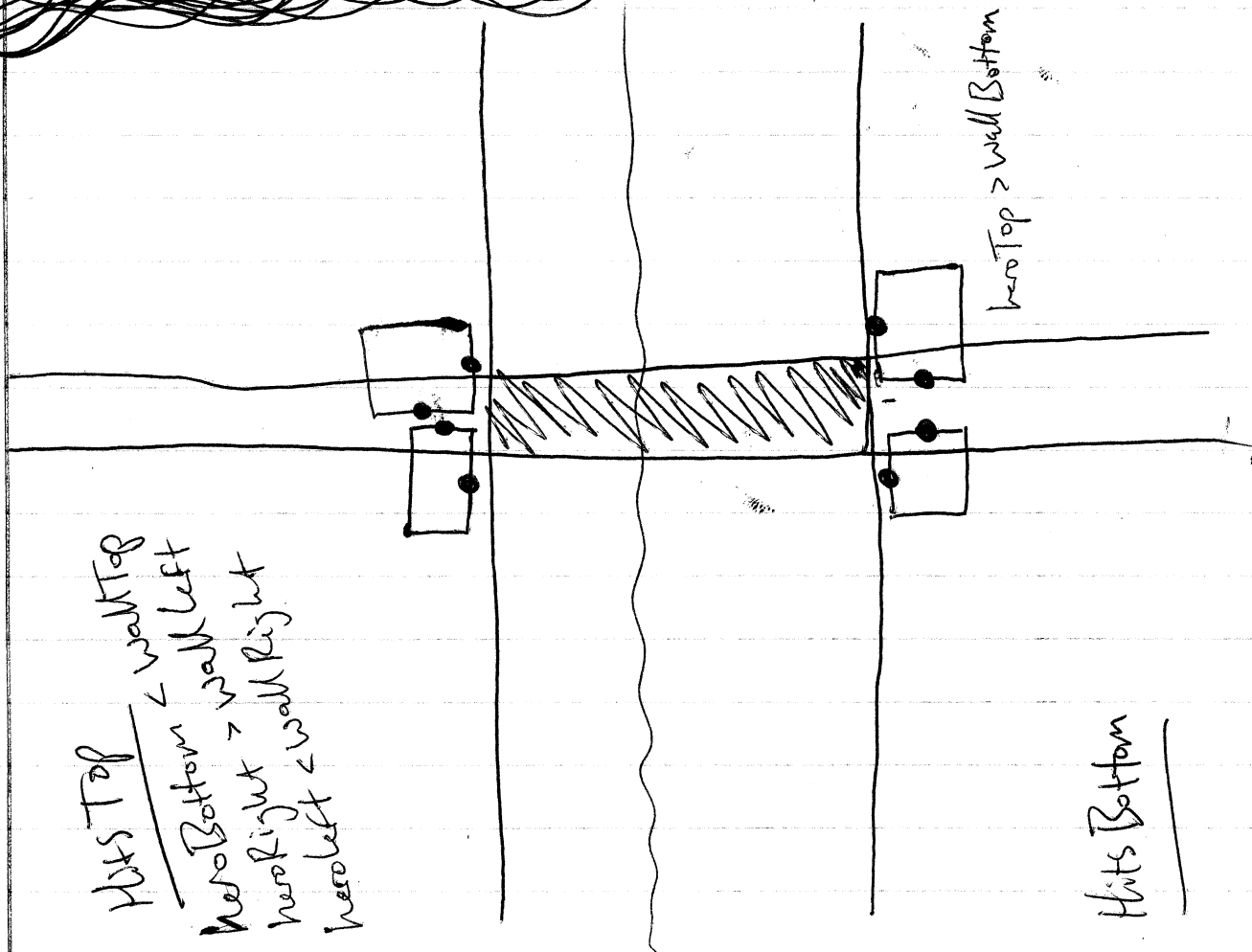
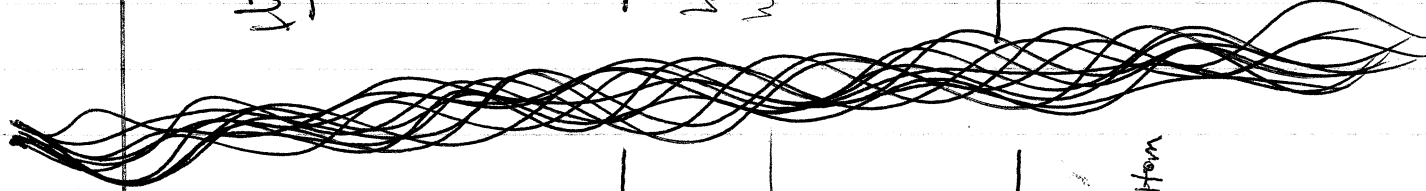
$heroBottom > wallTop$
 $heroTop < wallBottom$

$heroRight < wallLeft$

$heroLeft > wallRight$

$heroTop > wallBottom$

Hits Bottom



Build

Wall System ← make use of it!

— just make more levels

→ make a maze ← a maze in the dark