Name: _____

Student Workbook

Class: _____

BOOTSTRAP
www.bootstrapworld.org

Workbook v2.7

Brought to you the Bootstrap team:

- Emmanuel Schanzer
- Kathi Fisler
- Shriram Krishnamurthi
- Emma Youndtsmith
- Rosanna Sobota

Visual Design: Colleen Murphy

Bootstrap Units

01 Videogames and Coordinate Planes

02 Contracts, Strings, and Images

03 Intro to Definitions

04 Design Recipe

05 Game Animation

06 Comparing Functions

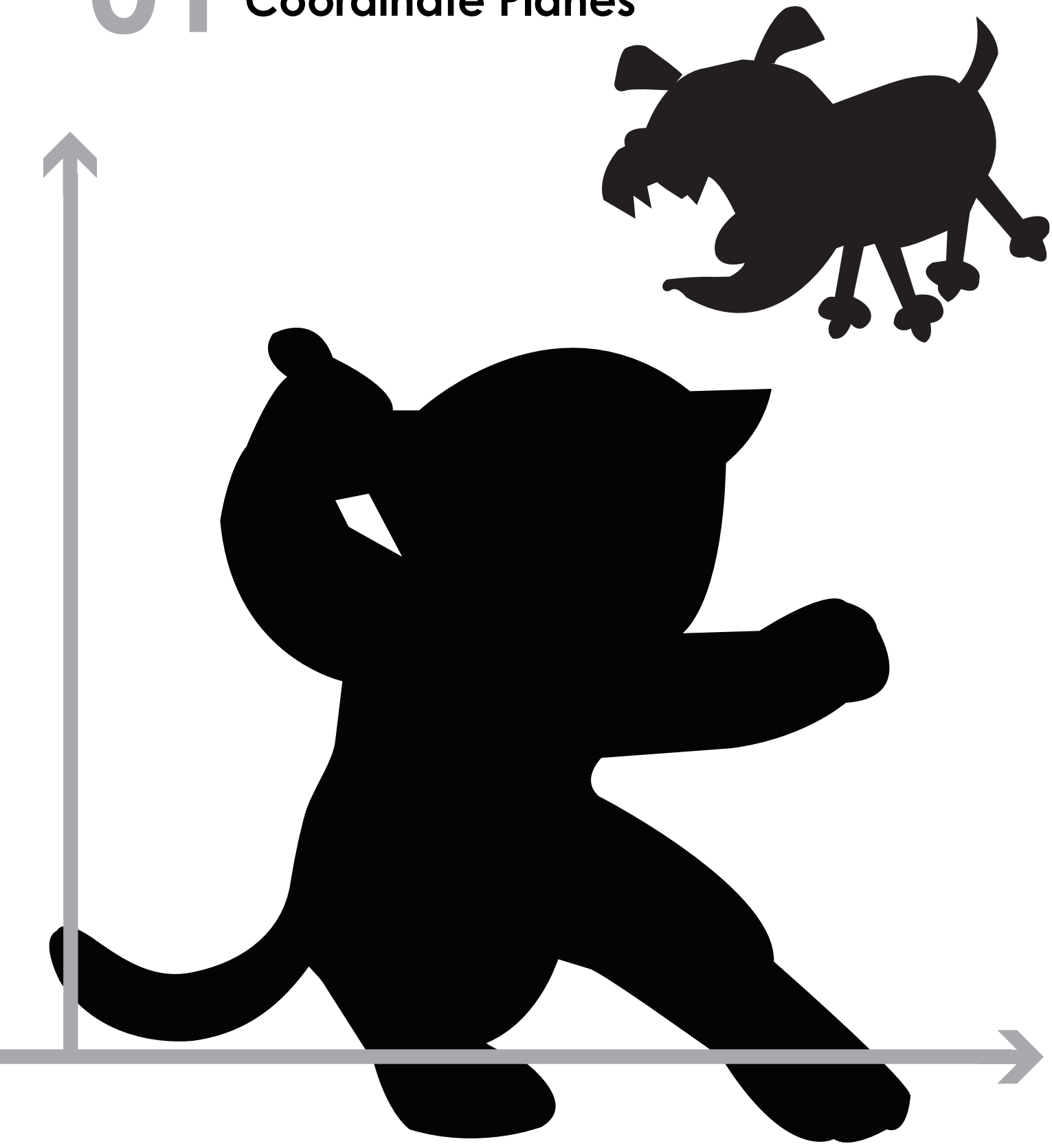
07 Conditional Branching

08 Collision Detection

09 Prepping for Launch

10 Additional Material

01 Videogames and Coordinate Planes



Lesson 1

Reverse-Engineering: How does NinjaCat work?

Thing in the game…	What changes about it?	More specifically…
cat	Position	x, y
ruby	position	x
clouds	position	x
dog	position	x
score	value	
background	nothing	

Finding Coordinates

The coordinates for the PLAYER (NinjaCat) are: (150 , 50)

x-coordinate *y-coordinate*

The coordinates for the DANGER (Dog) are: (450 , 50)

The coordinates for the TARGET (Ruby) are: (550 , 250)

Our Videogame

Created by (write your names): Jessica and James

Background

Our game takes place in: The Zoo

(space? the desert? a mall?)

The Player

The player is a Lion.

The player moves only up and down.

The Target

Your player GAINS points when they hit the target.

The Target is a Escaped gazelle.

The Target moves only to the left and right.

The Danger

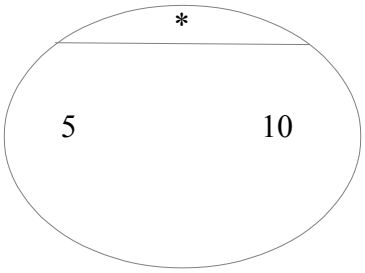
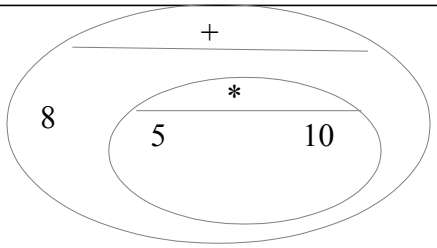
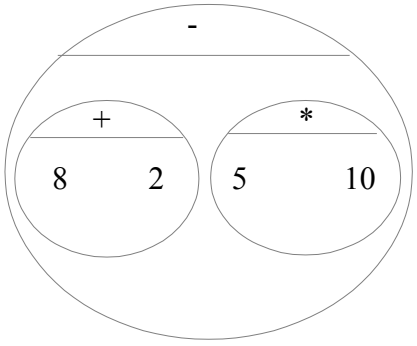
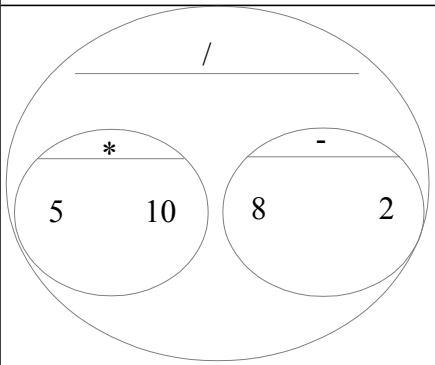
Your player LOSES points when they hit the danger.

The Danger is a Zookeeper.

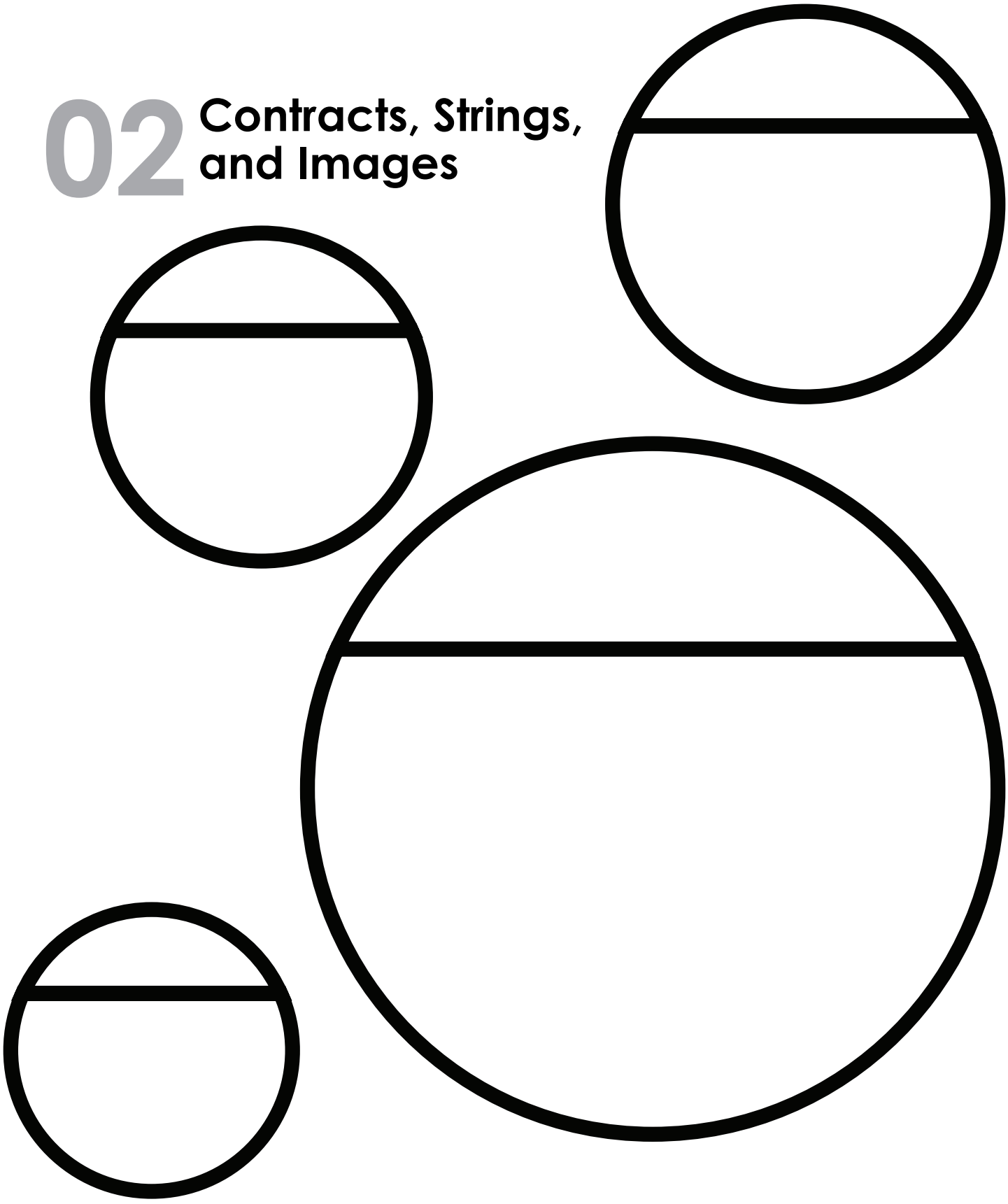
The Danger moves only to the left and right.

Circle of Evaluation Practice **Time: 5 minutes**

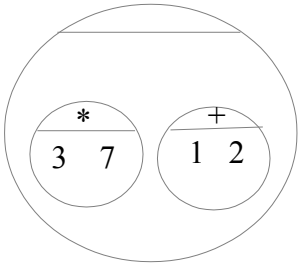
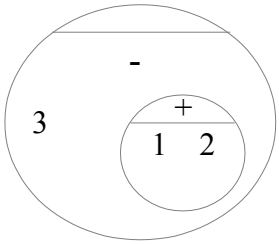
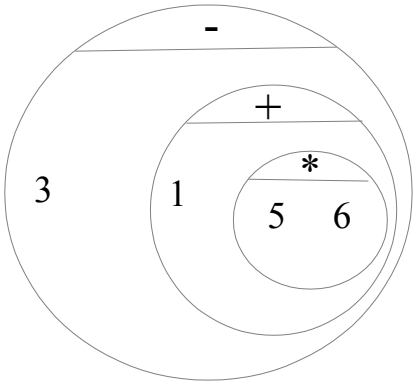
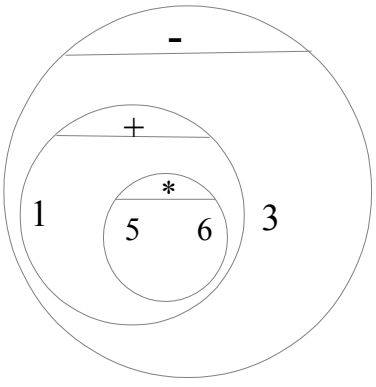
Don't forget to use the computer's symbols for things like multiply and divide!

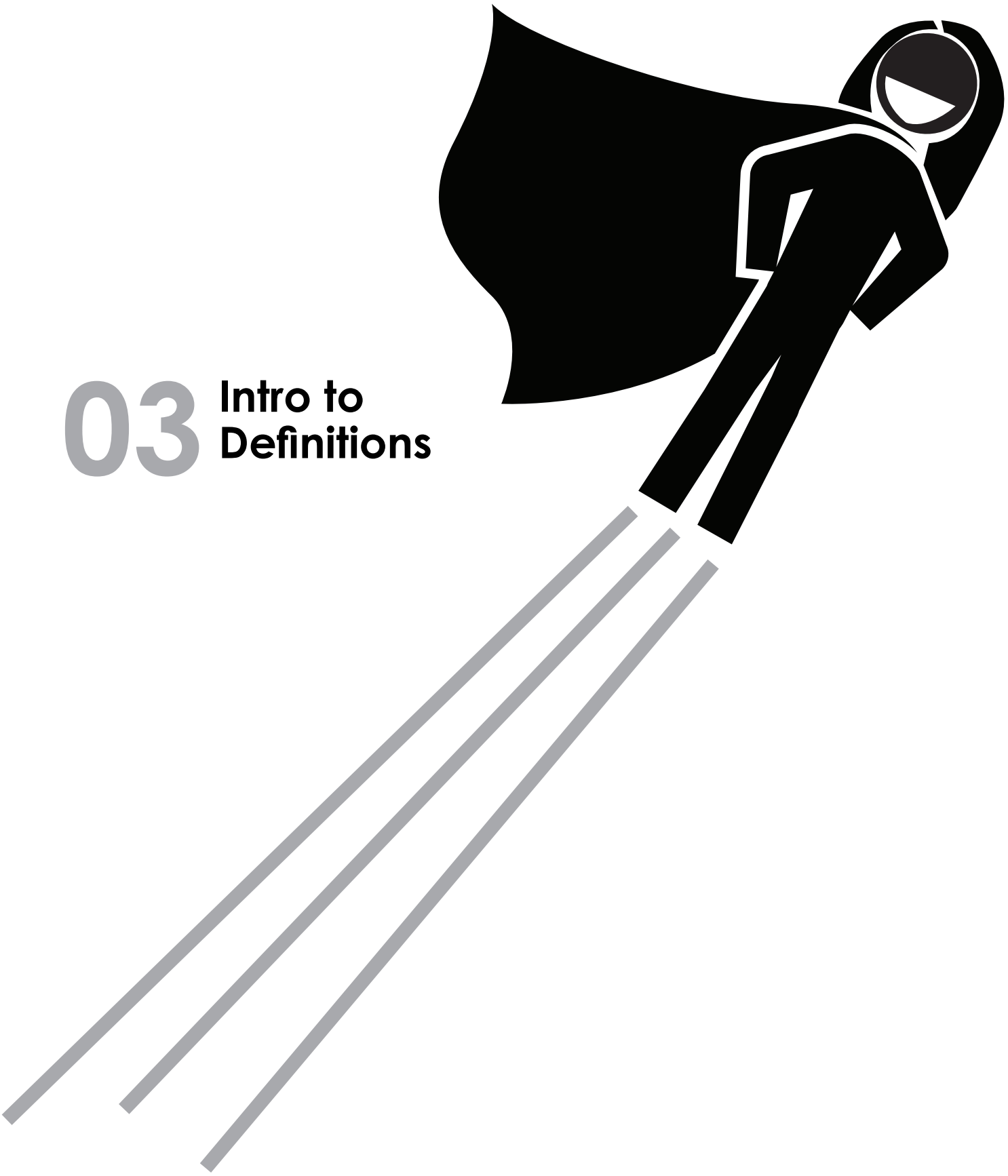
Math	***Circle of Evaluation***	***Racket Code***
5 x 10		(* 5 10)
8 + (5 x 10)		(+ 8 (* 5 10))
(8 + 2) - (5 x 10)		(- (+ 8 2) (* 5 10))
5 x 10 *8 - 2*		(/ (* 5 10) (- 8 2))

02 Contracts, Strings, and Images



Circles Competition

	Math	Round 1 -Circle of Evaluation	Round 2 - Racket Code
Challenge A	*(3 * 7) – (1 + 2)*		(- (* 3 7) (+ 1 2))
Challenge B	*3 - (1 + 2)*		(- 3 (+ 1 2))
Challenge C	*3 - (1 + (5 * 6))*		(- 3 (+ 1 (* 5 6)))
Challenge D	*(1 + (5 * 6)) - 3*		(- (+ 1 (* 5 6)) 3)



03 Intro to Definitions

Fast Functions

; gt : number -> image

name domain range

(EXAMPLE (gt 500) (triangle 500 "solid" "green"))

(EXAMPLE (gt 7) (triangle 7 "solid" "green"))

(define (gt size) (triangle size "solid" "green"))

; bc : number -> image

name domain range

(EXAMPLE (bc 19) (circle 19 "solid" "blue"))

(EXAMPLE (bc 43) (circle 43 "solid" "blue"))

(define (bc size) (circle size "solid" "blue"))

; double : number -> number

name domain range

(EXAMPLE (double 3) (* 2 3))

(EXAMPLE (double 9) (* 2 9))

(define (double num) (* 2 num))

; _____ : _____ -> _____

name domain range

(EXAMPLE (_____ _____) _____)

(EXAMPLE (_____ _____) _____)

(define (_____ _____) _____)

Fast Functions

; _____:_____ -> _____

name domain range

(EXAMPLE (_____ _____) _____)

(EXAMPLE (_____ _____) _____)

(define (_____ _____) _____)

; _____:_____ -> _____

name domain range

(EXAMPLE (_____ _____) _____)

(EXAMPLE (_____ _____) _____)

(define (_____ _____) _____)

; _____:_____ -> _____

name domain range

(EXAMPLE (_____ _____) _____)

(EXAMPLE (_____ _____) _____)

(define (_____ _____) _____)

; _____:_____ -> _____

name domain range

(EXAMPLE (_____ _____) _____)

(EXAMPLE (_____ _____) _____)

(define (_____ _____) _____)

04 Design Recipe

1 Contract

2 Example

3 Definition

Word Problem: rocket-height

Directions: *A rocket blasts off, traveling at 7 meters per second. Write a function called 'rocket-height' that takes in the number of seconds that have passed since the rocket took off, and which produces the height of the rocket at that time.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
; rocket-height : number → number
```

function name | *domain* | *range*

```
; Produce height of rocket after given number of seconds
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( rocket-height 0 ) (* 0 7) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( rocket-height 10 ) (* 10 7) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( rocket-height seconds )
```

function name | *variables*

```
        (* seconds 7) )
```

what the function does with those variables

Word Problem: lawn-area

Directions: *Use the Design Recipe to write a function 'lawn-area', which takes in the width and length of a lawn, and returns the area of the lawn. (Don't forget: area = length * width!)*

Contract and Purpose Statement ☐

Every contract has three parts...

```
;   lawn-area   :   number number   →   number
```

function name | *domain* | *range*

```
; Produce area of lawn with given length and width
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE(  lawn-area   10 20   )  (* 10 20)   )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE(  lawn-area   50 5   )  (* 50 5)   )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define(  lawn-area   length width  )
```

function name | *variables*

```
        (* length width)   )
```

what the function does with those variables

Word Problem: red-square

Directions: *Use the Design Recipe to write a function 'red-square', which takes in a number (the length of each side of the square) and outputs a solid red rectangle whose length and width are the same size.*

Contract and Purpose Statement ☐

Every contract has three parts...

; red-square : number → image

function name *domain* *range*

; Produce rectangle with given number as width and height

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

(EXAMPLE(red-square 4) (rectangle 4 4 "solid" "red"))

function name *input(s)* *what the function produces*

(EXAMPLE(red-square 12) (rectangle 12 12 "solid" "red"))

function name *input(s)* *what the function produces*

Definition ☐

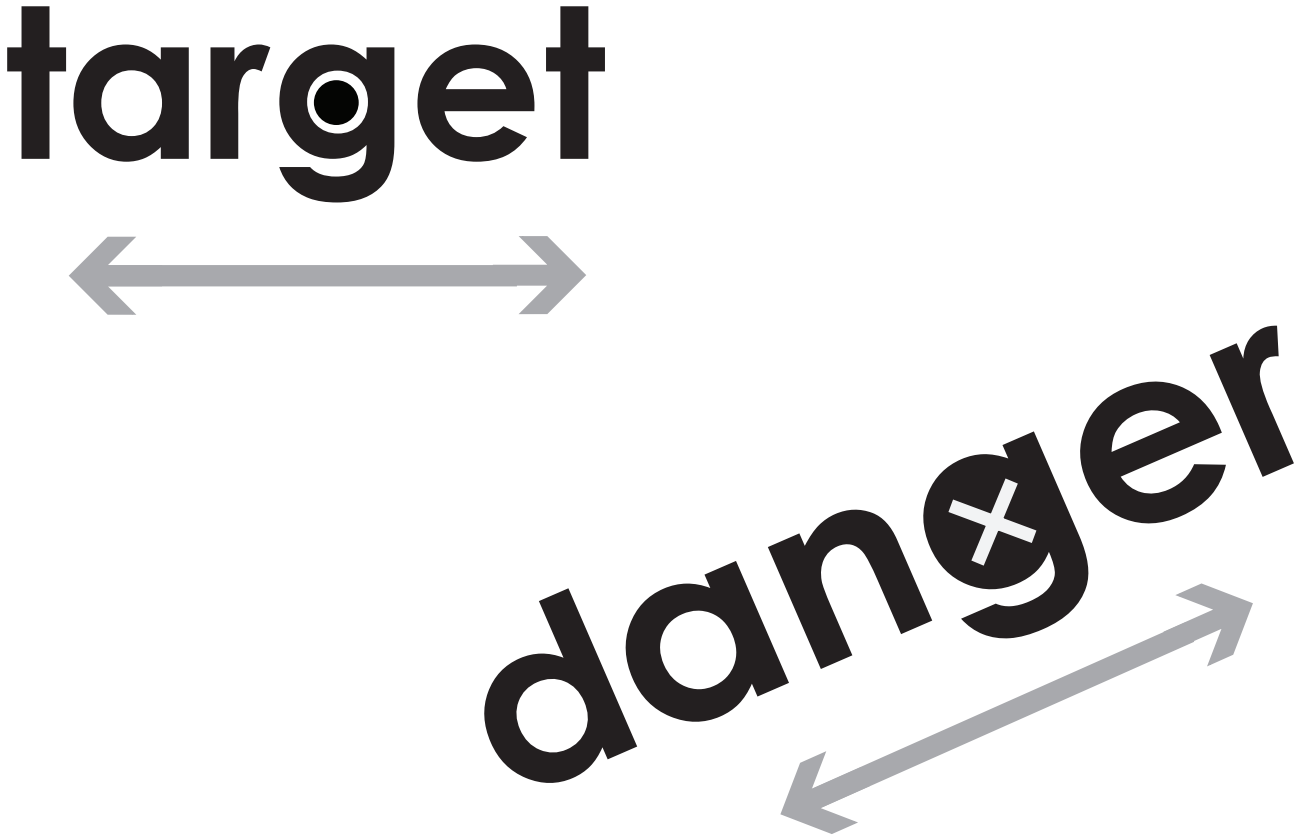
Write the definition, given variable names to all your input values...

(define(red-square size)

function name *variables*

(rectangle size size "solid" "red"))

what the function does with those variables



05 Game Animation

Word Problem: update-danger

Directions: *Use the Design Recipe to write a function 'update-danger', which takes in the danger's x-coordinate and produces the next x-coordinate, which is 50 pixels to the left.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
; update-danger : number → number
```

function name | *domain* | *range*

```
; Produce new coordinate by subtracting 50 from the given number
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( update-danger 100 ) (- 100 50) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( update-danger 35 ) (- 35 50) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( update-danger x )
```

function name | *variables*

```
        (- x 50) )
```

what the function does with those variables

Word Problem: update-target

Directions: *Write a function 'update-target', which takes in the target's x-coordinate and produces the next x-coordinate, which is 50 pixels to the right.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
; update-target : number → number
```

function name | *domain* | *range*

```
; Produce new coordinate by adding 50 to the given number
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( update-target 0 ) (+ 0 50) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( update-target 20 ) (+ 20 50) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( update-target x )
```

function name | *variables*

```
  (+ x 50) )
```

what the function does with those variables

“safe-left?”

06 Comparing Functions

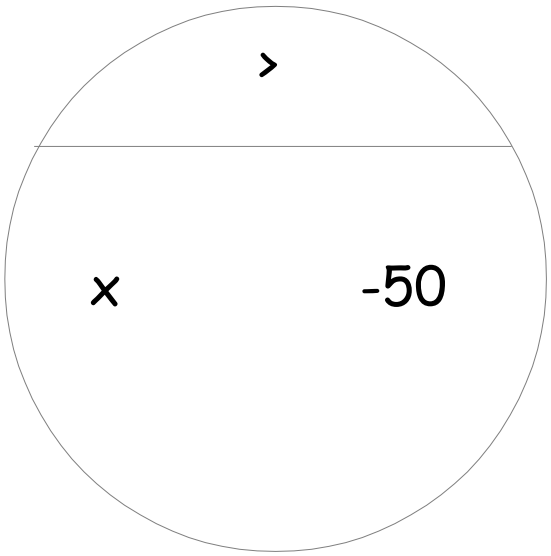
Protecting Sam

Sam is in a 640 x 480 yard. How far he can go to the left and right before he's out of sight?

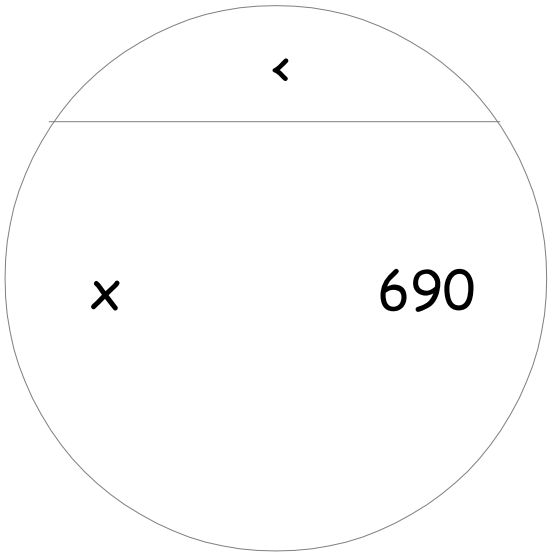
1. A piece of Sam is still visible on the left as long as... (> x -50)

2. A piece of Sam is still visible on the right as long as... (< x 690)

3. Draw the Circle of Evaluation for these two expressions in the circles below:



x -50



x 690

Word Problem: safe-left?

Directions: *Use the Design Recipe to write a function 'safe-left?', which takes in an x-coordinate and checks to see if it is greater than -50*

Contract and Purpose Statement ☐

Every contract has three parts...

```
;  safe-left?  :  number  →  boolean
```

function name | *domain* | *range*

```
; Determine whether given number is greater than -50
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( safe-left?  25  ) (> 25 -50) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( safe-left?  -51  ) (> -51 -50) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( safe-left?  x )
```

function name | *variables*

```
        (> x -50) )
```

what the function does with those variables

Word Problem: safe-right?

Directions: *Use the Design Recipe to write a function 'safe-right?', which takes in an x-coordinate and checks to see if it is less than 690.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
;  safe-right?  :  number  →  boolean
```

function name | *domain* | *range*

```
; Determine whether given number is less than 690
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( safe-right?  100  ) (< 100 690) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( safe-right?  820  ) (< 820 690) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( safe-right?  x )
```

function name | *variables*

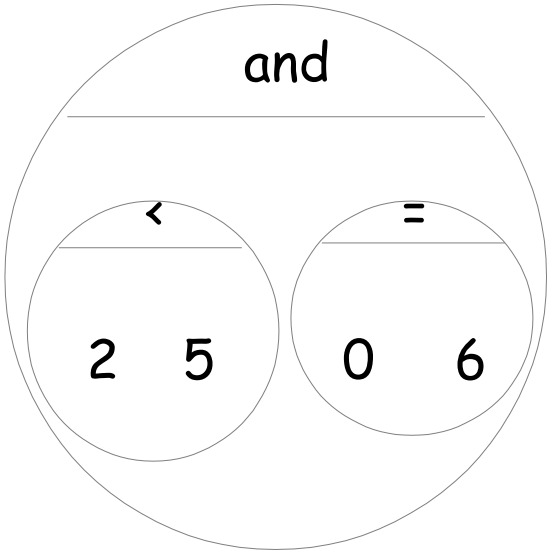
```
        (< x 690) )
```

what the function does with those variables

and / or

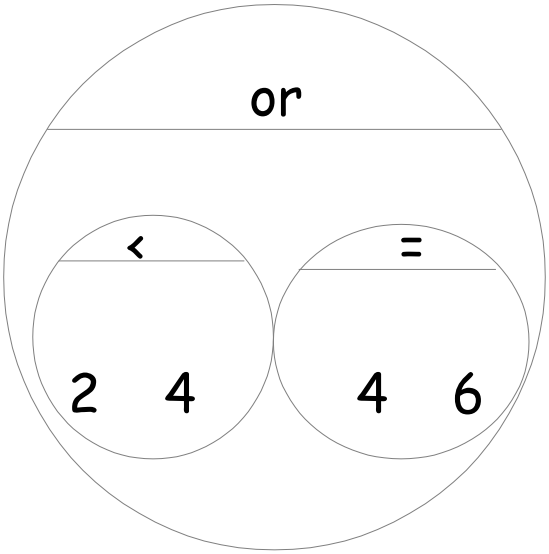
Write the Circles of Evaluation for these statements, and then convert them to Racket

1. Two is less than five, *and* zero is equal to six.



(and (< 2 5) (= 0 6))

2. Two is less than four *or* four is equal to six.



(or (< 2 4) (= 4 6))

Word Problem: onscreen?

Directions: *Use the Design Recipe to write a function 'onscreen?', which takes in the x-coordinate and checks to see if Sam is safe on the left AND safe on the right.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
;  onscreen?  :  number  →  boolean
```

function name | *domain* | *range*

```
; Sam is on the screen if his x-coordinate is safe on the left and safe
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE(  onscreen?  10  )
```

function name | *input(s)*

```
           (and (safe-left? 10) (safe-right? 10))  )
```

what the function produces

```
(EXAMPLE(  onscreen?  -15  )
```

function name | *input(s)*

```
           (and (safe-left? -15) (safe-right? -15))  )
```

what the function produces

Definition ☐

Write the definition, given variable names to all your input values...

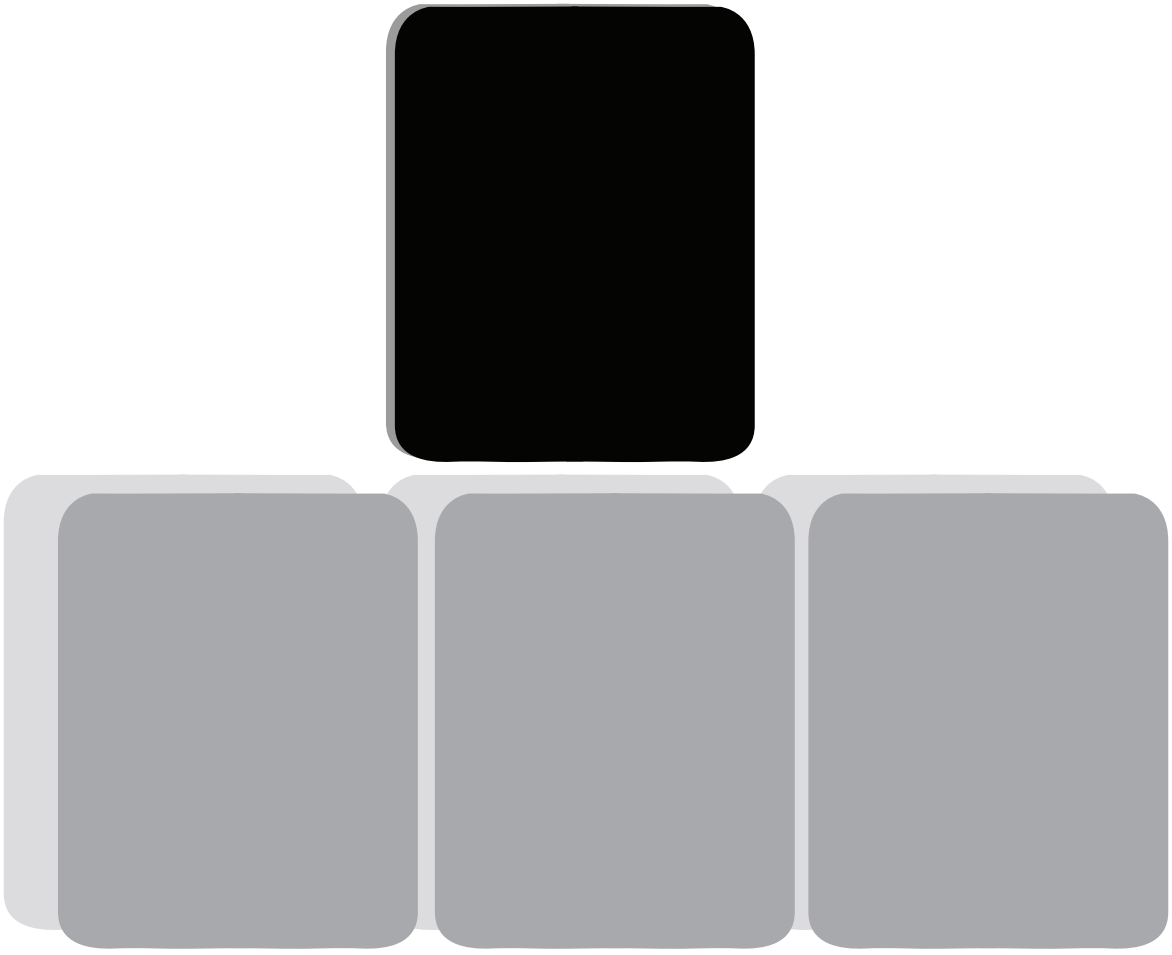
```
(define(  onscreen?  x  )
```

function name | *variables*

```
       (and (safe-left? x) (safe-right? x))  )
```

what the function does with those variables

07 Conditional Branching



Word Problem: cost

Directions: *Luigi's Pizza has hired you as a programmer. They offer Cheese ($9.00), Pepperoni ($10.50), Chicken ($11.25) and Broccoli ($10.25). Write a function called cost which takes in the name of a topping and outputs the price of a pizza with that topping.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
; cost : string → number
```

function name | *domain* | *range*

```
; Produce cost of a pizza with given topping
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( cost "cheese" ) 9 )
(EXAMPLE( cost "pepperoni" ) 10.5 )
(EXAMPLE( cost "chicken" ) 11.25 )
(EXAMPLE( cost "broccoli" ) 10.25 )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( cost topping )
        (cond
         [(string=? "cheese" topping) 9 ]
         [(string=? "pepperoni" topping) 10.5 ]
         [(string=? "chicken" topping) 11.25 ]
         [(string=? "broccoli" topping) 10.25 ]
         [else: 1000000 ]))
```

function name | *variables*

Word Problem: update-player

Directions: *Write a function called update-player, which takes in the player's y-coordinate and the name of the key pressed, and returns the new y-coordinate.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
; update-player : number string → number
```

function name | *domain* | *range*

```
; Produce new y-coordinate depending on key press
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( update-player   320  "up"    )  (+ 320 20)  )
(EXAMPLE( update-player   100  "up"    )  (+ 100 20)  )
(EXAMPLE( update-player   320  "down"  )  (- 320 20)  )
(EXAMPLE( update-player   100  "down"  )  (- 100 20)  )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( update-player   y  key  )
```

function name | *variables*

```
        (cond

         [(string=? "up" key)          (+ y 20)  ]

         [(string=? "down" key)        (- y 20)  ]

         [else:      y                           ]))
```

08 Collision Detection

collision

distance

Word Problem: line-length

Directions: *Write a function called 'line-length', which takes in two numbers and returns the *positive difference* between them. It should always subtract the smaller number from the bigger one, and if they are equal it should return zero.*

Contract and Purpose Statement ☐

Every contract has three parts...

```
;  line-length  :  number number  →  number
```

function name | *domain* | *range*

```
;  Produce positive difference between two given numbers
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( line-length  10 5  ) (- 10 5) )
```

function name | *input(s)* | *what the function produces*

```
(EXAMPLE( line-length  2 8  ) (- 8 2) )
```

function name | *input(s)* | *what the function produces*

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( line-length  a b  )
```

function name | *variables*

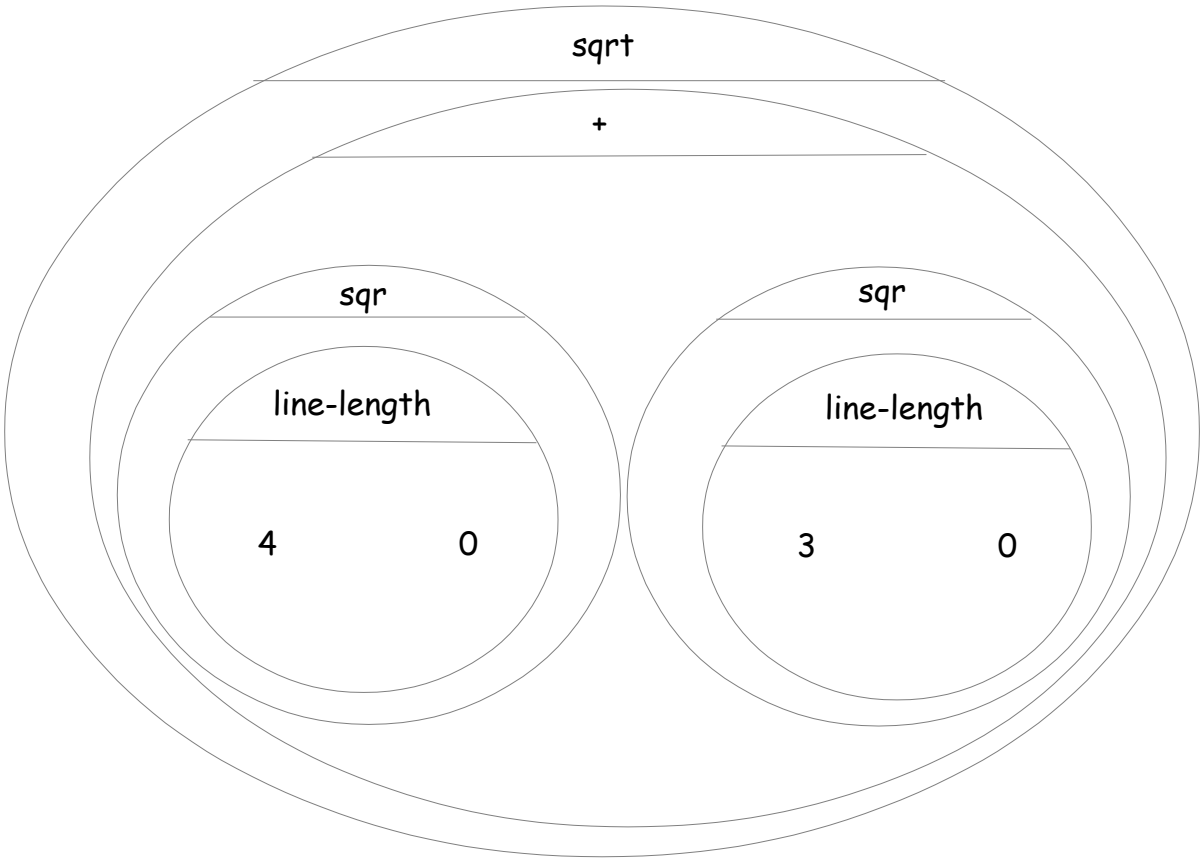
```
        (cond

         [(> a b)  (- a b)  ]

         [(< a b)  (- b a)  ]))
```

The Distance Formula (an example)

The distance between the points (0, 0) and (4, 3) is given by:

$$\sqrt{(line\text{-}length\ \ 4\ \ 0)^2 + \ (line\text{-}length\ \ 3\ \ 0)^2}$$

Convert the formula above into a Circle of Evaluation (We've already gotten you started!)



Convert the Circle of Evaluation to code, then label the numbers with (x1,y1) & (y1,y2):

```
(sqrt (+   (sqr (line-length 4 0))
           (sqr (line-length 3 0)))))
```

Word Problem: distance

Directions: *Write a function distance, which takes FOUR inputs:*

- *px: The x-coordinate of the player*
- *py: The y-coordinate of the player*
- *cx: the x-coordinate of another game character*
- *cy: the y-coordinate of another game character*

It should return the distance between the two, using the Distance formula. (HINT: look at what you did on the previous page!)

Contract and Purpose Statement ☐

Every contract has three parts...

```
;    distance    :    number number number number    →    number
```
function name | *domain* | *range*

```
; Produce distance between two points with given coordinates
```
what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE(   distance    0 0 3 4   )
```
function name | *input(s)*

```
(sqrt (+ (sq (line-length 3 0)) (sq (line-length 4 0))))
```
what the function produces

```
(EXAMPLE(   distance    10 20 13 24   )
```
function name | *input(s)*

```
(sqrt (+ (sq (line-length 13 10)) (sq (line-length 24 20)))))
```
what the function produces

Definition ☐

Write the definition, given variable names to all your input values...

```
(define(   distance    px py cx cy   )
```
function name | *variables*

```
(sqrt (+ (sqr (line-length px cx)) (sqr (line-length py cy))))
```
what the function does with those variables

Word Problem: collide?

Directions: *Write a function collide?, which takes FOUR inputs:*

- *px: The x-coordinate of the player*
- *py: The y-coordinate of the player*
- *cx: the x-coordinate of another game character*
- *cy: the y-coordinate of another game character*

Are the coordinates of the player within 50 pixels of the coordinates of the other character?

Contract and Purpose Statement ☐

Every contract has three parts...

```
;    collide?    :    number number number number    →    boolean
```

function name | *domain* | *range*

```
; Is the Player (px, py) within 50 pixels of another Character (cx, cy)?
```

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE(   collide?    25 75 30 270 )
```

function name | *input(s)*

```
          (< (distance 25 75 30 270) 50)                    )
```

what the function produces

```
(EXAMPLE(   collide?    25 75 100 75 )
```

function name | *input(s)*

```
          (< (distance 25 75 100 75) 50)                    )
```

what the function produces

Definition ☐

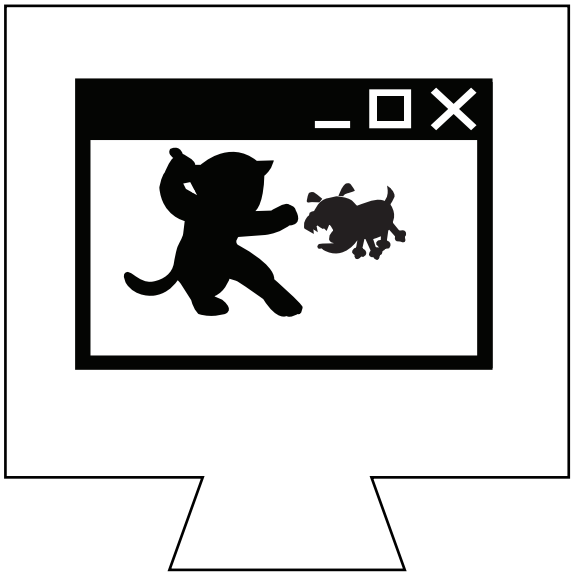
Write the definition, given variable names to all your input values...

```
(define(   collide?    px py cx cy )
```

function name | *variables*

```
       (< (distance px py cx cy) 50)                        )
```

what the function does with those variables



09 Presentation Preparation

Lesson 9

Catchy Intro: Feel like you never get enough to eat? So does Leo. Come catch your prey, and escape the zookeeper!

Name, Age, Grade: Jessica Programmer, 12, 7th grade

Game Title: Run for your Supper

Back Story:One day, a young lion was sitting in his cage. He saw an escaped gazelle come running past. It was lunch time, and he was hungry, so he leapt out to catch food. He has to run fast to grab food and escape the evil zookeeper.

Characters: Player: Leo the lion.

Danger: Zoe Zookeeper.

Target: Gary Gazelle

Explain a piece of your code: My update-danger function takes in the current x coordinate of the gazelle, and adds 50 to it. This moves the gazelle 50 pixels to the right.

Presentation Feedback

For each question, circle the answer that fits best.

Was the introduction catchy?	No way!	A little.	Definitely!
Did they talk about their characters?	No way!	A little.	Definitely!
Did they explain the code well?	No way!	A little.	Definitely!
Did they speak slowly enough?	No way!	A little.	Definitely!
Did they speak loudly enough?	No way!	A little.	Definitely!
Were they standing confidently?	No way!	A little.	Definitely!
Did they make eye contact?	No way!	A little.	Definitely!

Presentation Feedback

For each question, circle the answer that fits best.

Was the introduction catchy?	No way!	A little.	Definitely!
Did they talk about their characters?	No way!	A little.	Definitely!
Did they explain the code well?	No way!	A little.	Definitely!
Did they speak slowly enough?	No way!	A little.	Definitely!
Did they speak loudly enough?	No way!	A little.	Definitely!
Were they standing confidently?	No way!	A little.	Definitely!
Did they make eye contact?	No way!	A little.	Definitely!

Word Problem: red-shape

Directions: *Write a function called red-shape, which takes in the name of a shape and draws that shape (solid and red). Add an else clause that produces a sensible output.*

Contract and Purpose Statement ☐

Every contract has three parts...

; red-shape : string → image

function name | domain | range

; Create a solid red shape of the given kind

what does the function do?

Examples ☐

Write some examples, then circle and label what changes...

```
(EXAMPLE( red-shape "circle" ) (circle 50 "solid" "red") )
```

function name | input(s) | what the function produces

```
(EXAMPLE( red-shape "triangle" ) (triangle 50 "solid" "red") )
```

function name | input(s) | what the function produces

```
(EXAMPLE( red-shape "rectangle" ) (rectangle 99 9 "solid" "red") )
```

function name | input(s) | what the function produces

```
(EXAMPLE( red-shape "star" ) (triangle 50 "solid" "red") )
```

function name | input(s) | what the function produces

Definition ☐

Write the definition, given variable names to all your input values...

```
(define( red-shape shape )
```

function name | variables

```
    (cond
     [(string=? "circle" shape)      (circle 50 "solid" "red")              ]
     [(string=? "triangle" shape)    (triangle 50 "solid" "red")            ]
     [(string=? "rectangle" shape)   (rectangle 99 9 "solid" "red")         ]
     [(string=? "star" shape)        (star 50 "solid" "red")                ]
     [else: (text 20 "???" "red")                                          ]))
```

Translating into Algebra

Value Definitions

Racket Code	Algebra
`(define x 10)`	x = 10
`(define y (* x 2))`	y = x*2
`(define z (+ x y))`	**z = x + y**
`(define age 14)`	**age = 14**
`(define months (* age 12))`	**months = age * 12**
`(define days (* months 30))`	**days = months * 30**
`(define hours (* days 24))`	**hours = days * 24**
`(define minutes (* hours 60))`	**minutes = hours * 60**

Function Definitions

Racket Code	Algebra
`(define (area length width)` `  (* length width))`	area(length, width) = length * width
`(define (circle-area radius)` `  (* pi (sqr radius)))`	**circle-area(radius) = pi * radius^2**

`(define (distance x1 y1 x2 y2)`
`  (sqrt (+ (sqr (- x1 x2))`
`  (sqr (- y1 y2)))))`
	distance(x1, y1, x2, y2)= $\sqrt{(x1-x2)^2+(y1-y2)^2}$

Design Recipe

A rocket is flying from Earth to Mars at 80 miles per second. Write a function that describes the **distance** *D* that the rocket has traveled, as a function of **time** *t*.

I. Contract+Purpose Statement

Every contract has three parts:

; D : Number -> Number

name Domain Range

; Given the number of seconds, produce the height of the rocket if it moves at 80mi/sec

What does the function do?

II. Give Examples

Write an example of your function for some sample inputs

D(1) = 80 * 1

Use the function here What should the function produce?

D(2) = 80 * 2

Use the function here What should the function produce?

D(3) = 80 * 3

Use the function here What should the function produce?

D(4) = 80 * 4

Use the function here What should the function produce?

III. Definition

Write the formula, giving variable names to all your input values.

D(time) = 80 * time

Design Recipe

A rocket is traveling from Earth to Mars at 80 miles per second. Write a function that describes the ***time*** the rocket has been traveling, as a function of ***distance***.

I. Contract+Purpose Statement

Every contract has three parts:

; **time** : **Number** -> **Number**

name | Domain | Range

; Given the distance, produce the time-traveled if it moves at 80mi/sec

What does the function do?

II. Give ExamplesGive Examples

Write an example of your function for some sample inputs

time(0) = 0/80

Use the function here | What should the function produce?

time(10) = 10/80

Use the function here | What should the function produce?

time(80) = 80/80

Use the function here | What should the function produce?

time(190) = 190/80

Use the function here | What should the function produce?

III. Definition

Write the Formula, giving variable names to all your input values.

time(distance) = distance/80

Design Recipe

A rocket leaves Earth, headed for Mars at 80 miles per second. **At the exact same time**, an asteroid leaves Mars traveling towards Earth, moving at 70 miles per second. If the distance from the Earth to Mars is 50,000,000 miles, how long will it take for them to meet?

I. Contract+Purpose Statement

Every contract has three parts:

; **collide** : **Number** -> **Number**

name Domain Range

; Given the distance between a rocket (moving at 80mi/sec) & asteroid (70mi/sec), when will they collide?

What does the function do?

II. Give ExamplesGive Examples

Write an example of your function for some sample inputs

collide(0) = 0/150

Use the function here What should the function produce?

collide(150) = 150/150

Use the function here What should the function produce?

collide(700) = 700/150

Use the function here What should the function produce?

collide(50,000,000) = 50,000,000/150

Use the function here What should the function produce?

III. Definition

Write the Formula, giving variable names to all your input values.

collide(distance-between) = distance-between/150

Design Recipe

I. Contract+Purpose Statement

Every contract has three parts:

; _____ : _____ -> _____

name Domain Range

; _____

What does the function do?

II. Give Examples

Write an example of your function for some sample inputs

	=	
Use the function here		What should the function produce?

	=	
Use the function here		What should the function produce?

	=	
Use the function here		What should the function produce?

	=	
Use the function here		What should the function produce?

III. Definition

Write the Formula, giving variable names to all your input values.

=

Design Recipe

I. Contract+Purpose Statement

Every contract has three parts:

; _____ : _____ -> _____

name Domain Range

; _____

What does the function do?

II. Give Examples

Write an example of your function for some sample inputs

_____ =

Use the function here What should the function produce?

_____ =

Use the function here What should the function produce?

_____ =

Use the function here What should the function produce?

_____ =

Use the function here What should the function produce?

III. Definition

Write the Formula, giving variable names to all your input values.

_____ =

Contracts

Name	Domain	Range	example
; +	: Number Number	→Number	(+ 3 4)
; -	: Number Number	→Number	(- 3 4)
; *	: Number Number	→Number	(* 3 4)
; /	: Number Number	→Number	(/ 3 4)
; sqr	: Number	→Number	(sqr 10)
; sqrt	: Number	→Number	(sqrt 10)
; star	: Number String String	→ Image	(star 50 “solid” “red”)
; circle	: Number String String	→Image	(circle 50 “solid” “red”)
; triangle	: Number String String	→Image	(triangle 50 “solid” “red”)
; rectangle	: Number Number String String	→Image	(rectangle 20 100 “solid” “red”)
; ellipse	: Number Number String String	→Image	(ellipse 75 150 “solid” “red”)
; text	: String Number String	→ Image	(text “hello world” 50 “blue”)
; rotate	: Number Image	→ Image	(rotate 45 (triangle 20 “solid” “blue”))
; scale	: Number Image	→ Image	(scale 2 (star 20 “solid” “yellow”))
; bitmap/url	: String	→ Image	(bitmap/url “www…”)
; =	: Number Number	→Boolean	(= 3 4)
; >	: Number Number	→Boolean	(> 3 4)

Contracts

Name	Domain	Range	example
; <	: Number Number	→ Boolean	(< 3 4)
; string=?	: String String	→ Boolean	(string=? “apples” “oranges”)
; and	: Boolean Boolean	→ Boolean	(and (< 3 1) (= 6 4))
; or	: Boolean Boolean	→ Boolean	(or (string=? “hello” “hello”) (> 2 6))
; *put-image*	: *Image Number Number Image*	→ *Image*	*(put-image (circle ……) 30 50 (rectangle…))*
; >=	: *Number Number*	→ *Boolean*	*(>= 1 2)*
; <=	: *Number Number*	→ *Boolean*	*(<= (+ 3 1) 4)*
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	
;	:	→	