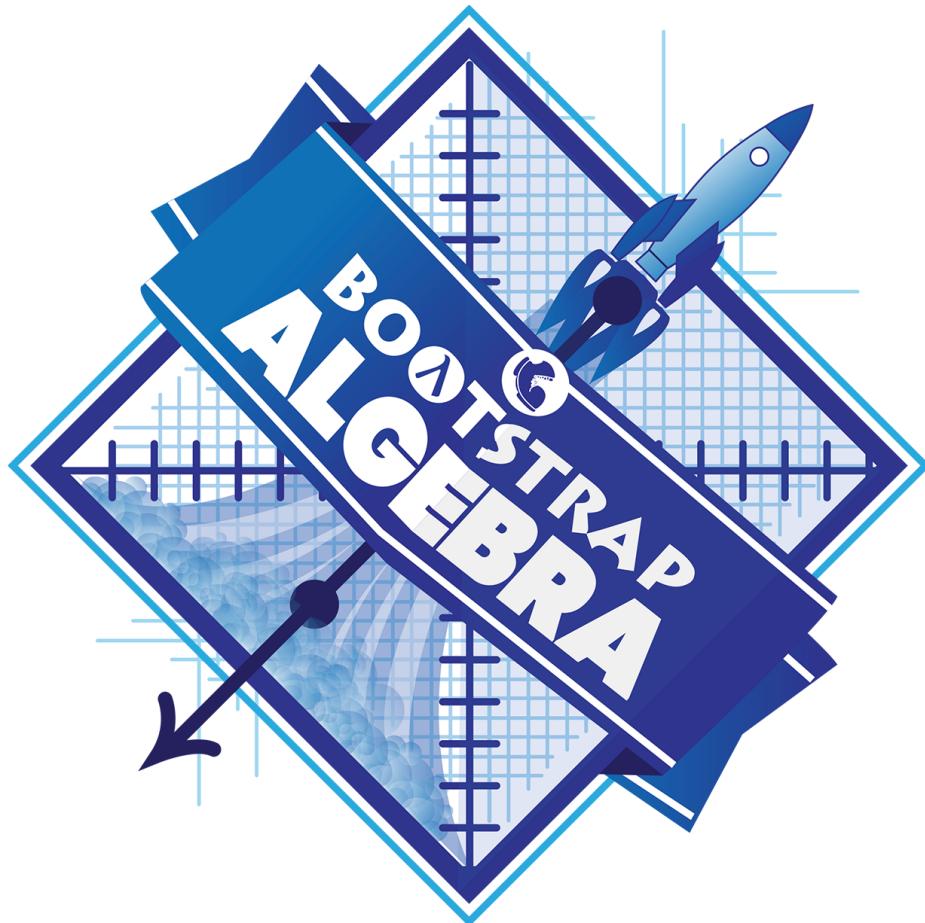


Name: \_\_\_\_\_



## Student Workbook

Class: \_\_\_\_\_



Workbook v2.7

Brought to you the Bootstrap team:

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

Visual Design: Colleen Murphy

---

Bootstrap is licensed under a Creative Commons 3.0 Unported License. Based on a work from [www.BootstrapWorld.org](http://www.BootstrapWorld.org). Permissions beyond the scope of this license may be available at [schanzer@BootstrapWorld.org](mailto:schanzer@BootstrapWorld.org).

# Bootstrap Units

<b>01</b>	Videogames and Coordinate Planes	<b>06</b>	Comparing Functions
<b>02</b>	Contracts, Strings, and Images	<b>07</b>	Conditional Branching
<b>03</b>	Intro to Definitions	<b>08</b>	Collision Detection
<b>04</b>	Design Recipe	<b>09</b>	Prepping for Launch
<b>05</b>	Game Animation	<b>10</b>	Additional Material

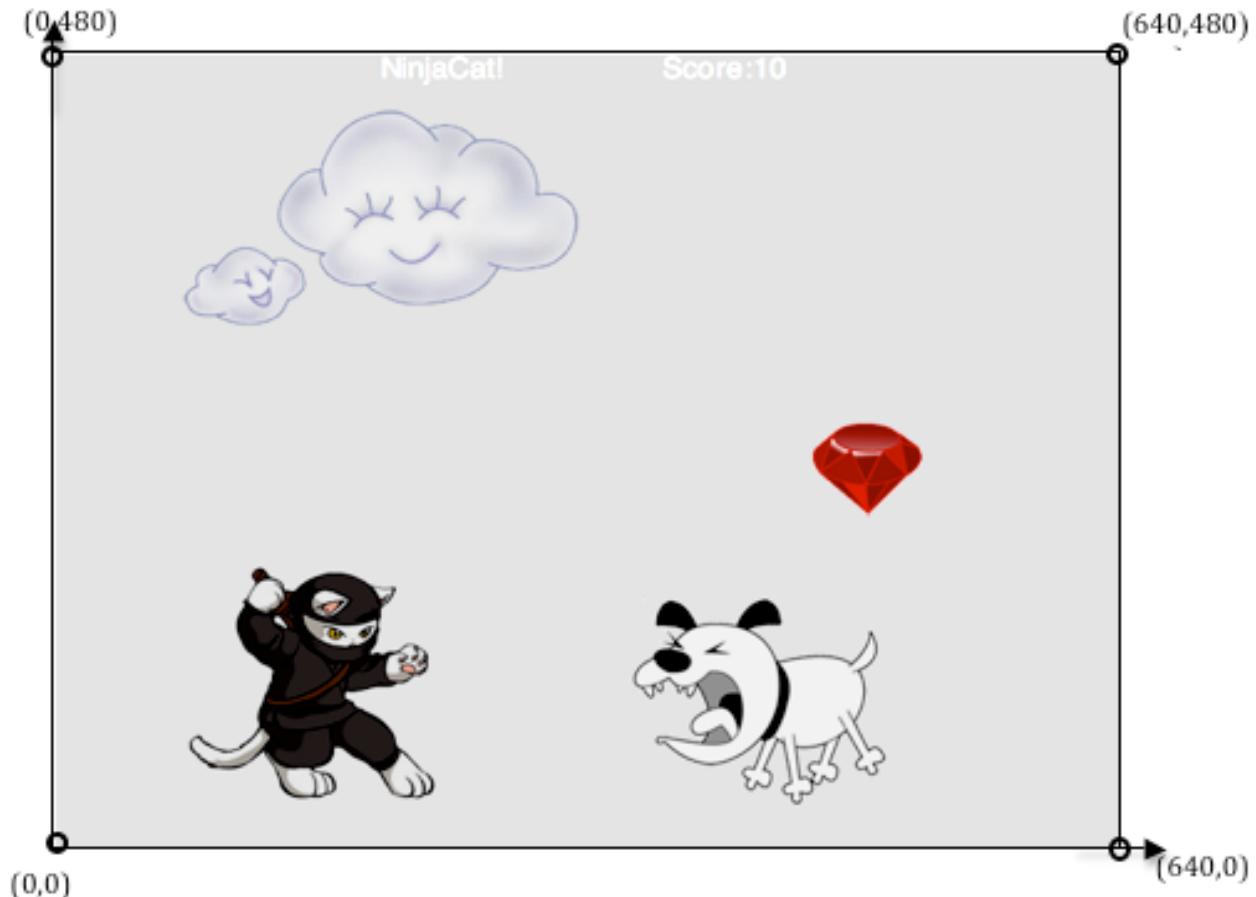
# 01 Videogames and Coordinate Planes



# Lesson 1

Reverse-Engineering: How does NinjaCat work?

## Finding Coordinates



The coordinates for the PLAYER (NinjaCat) are:  $(\text{x-coordinate}, \text{y-coordinate})$

The coordinates for the DANGER (Dog) are:  $(\text{x-coordinate}, \text{y-coordinate})$

The coordinates for the TARGET (Ruby) are:  $(\text{x-coordinate}, \text{y-coordinate})$

# Our Videogame

Created by (write your names): \_\_\_\_\_

## Background

Our game takes place in: \_\_\_\_\_  
(space? the desert? a mall?)

## The Player

*The player is a \_\_\_\_\_.*

The player moves only up and down.

## The Target

*Your player GAINS points when they hit the target.*

*The Target is a \_\_\_\_\_.*

The Target moves only to the left and right.

## The Danger

*Your player LOSES points when they hit the danger.*

*The Danger is a \_\_\_\_\_.*

The Danger moves only to the left and right.

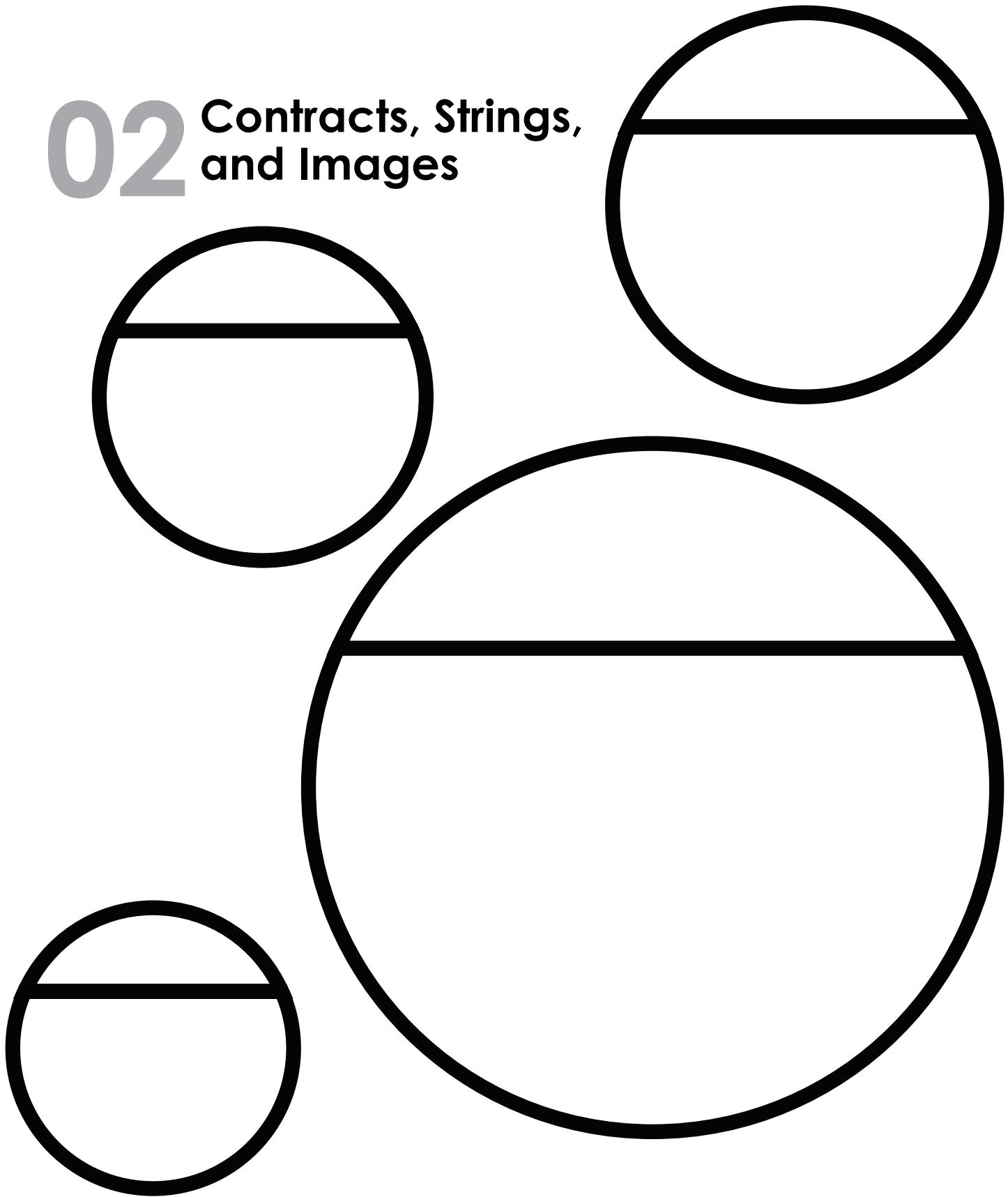
## Circle of Evaluation Practice

Time: 5 minutes

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

Math	Circle of Evaluation	Racket Code
$5 \times 10$		
$8 + (5 \times 10)$		
$(8 + 2) - (5 \times 10)$		
$\frac{5 \times 10}{8 - 2}$		

## 02 Contracts, Strings, and Images



## Circles Competition

Time: 5 minutes

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

# 03

## Intro to Definitions



## 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 (\_\_\_\_\_ \_\_\_\_\_) \_\_\_\_\_)

## 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

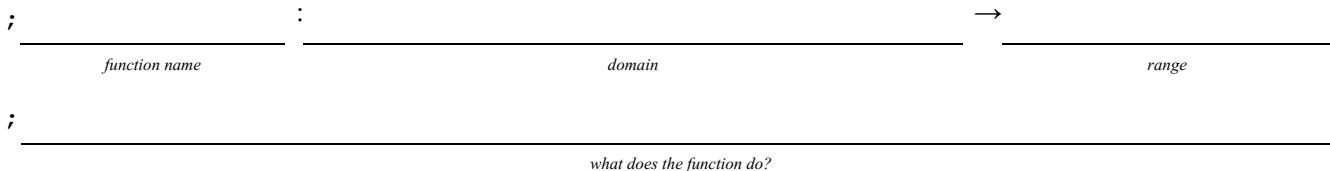


## Word Problem: rocket-height

**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.

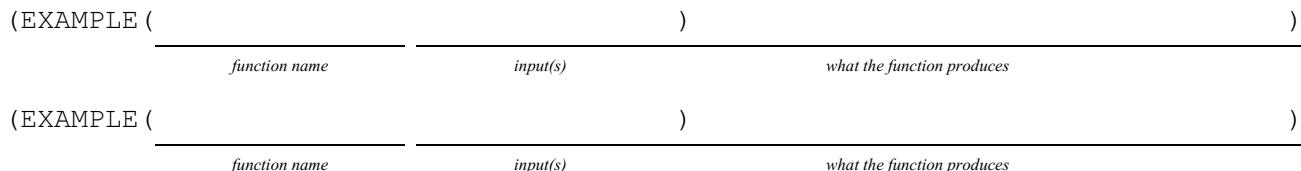
### Contract and Purpose Statement

Every contract has three parts...



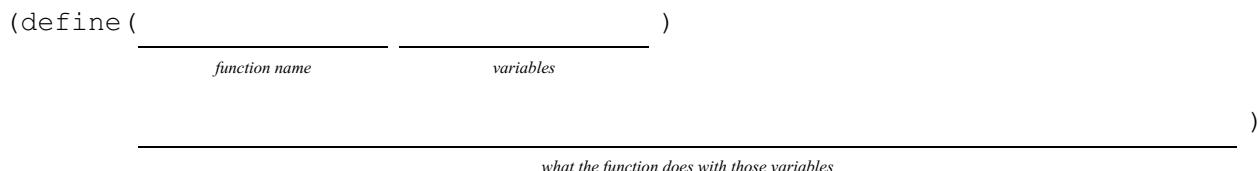
### Examples

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



### Definition

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

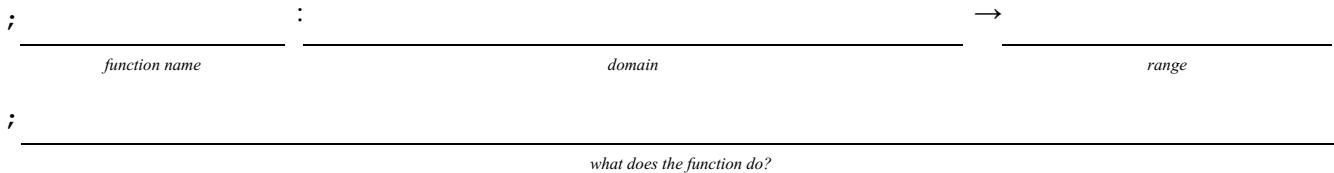


## Word Problem: lawn-area

**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!)

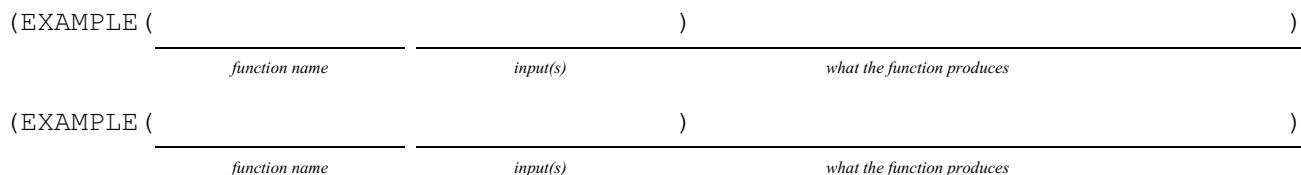
### Contract and Purpose Statement

Every contract has three parts...



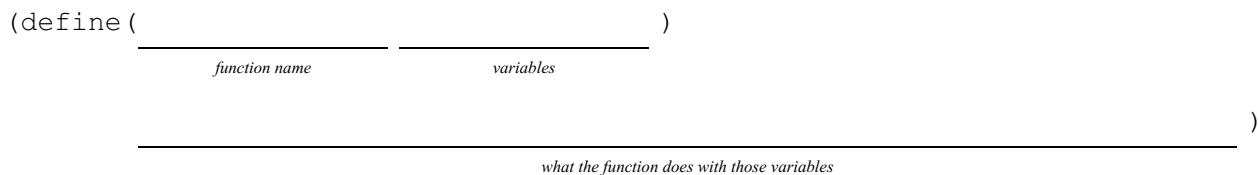
### Examples

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



### Definition

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



## 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...*

Diagram illustrating the components of a function definition:

The diagram shows a horizontal line divided into three segments by vertical bars. The first segment is labeled *function name*, the second is labeled *domain*, and the third is labeled *range*. An arrow points from the *range* segment to the right. Below the line, the text *what does the function do?* is centered.

## Examples

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

(EXAMPLE ( \_\_\_\_\_ ) )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
*function name*            *input(s)*            *what the function produces*

(EXAMPLE ( \_\_\_\_\_ ) )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
*function name*           *input(s)*           *what the function produces*

## Definition

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

```
(define (function name variables)  
  what the function does with those variables)
```

# target



# danger



## 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...*

## Examples

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

(EXAMPLE ( \_\_\_\_\_ ) )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
*function name*           *input(s)*           *what the function produces*

(EXAMPLE ( \_\_\_\_\_ ) )  
                  *function name*      *input(s)*      *what the function produces*

## Definition

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

```
(define (function name variables)  
  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...*

;

:

---

*function name*

*domain*

$\rightarrow$

---

*range*

;

---

*what does the function do?*

## Examples

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

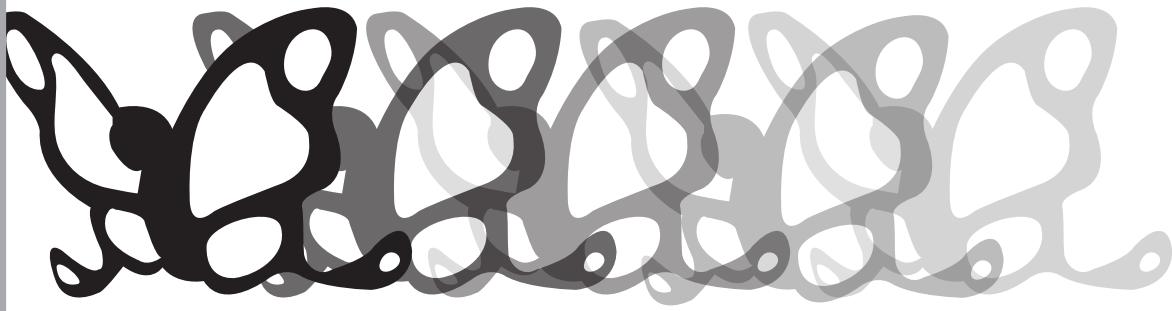
(EXAMPLE ( \_\_\_\_\_ ) )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
*function name*            *input(s)*            *what the function produces*

(EXAMPLE ( \_\_\_\_\_ ) )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_

## Definition

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

```
(define (function name variables)  
  what the function does with those variables)
```



**“safe-left?”**

## 06 Comparing Functions

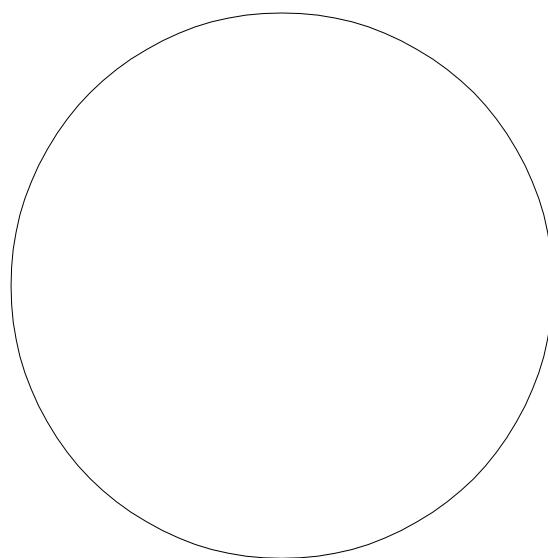
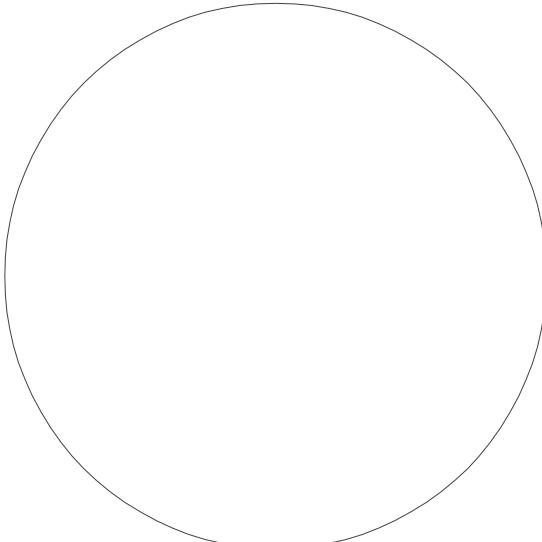
## Sam the Butterfly

Sam is in a  $640 \times 480$  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...  $\underline{(> \quad x \quad -50)}$

2. A piece of Sam is still visible on the right as long as...  $\underline{\hspace{2cm}}$

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

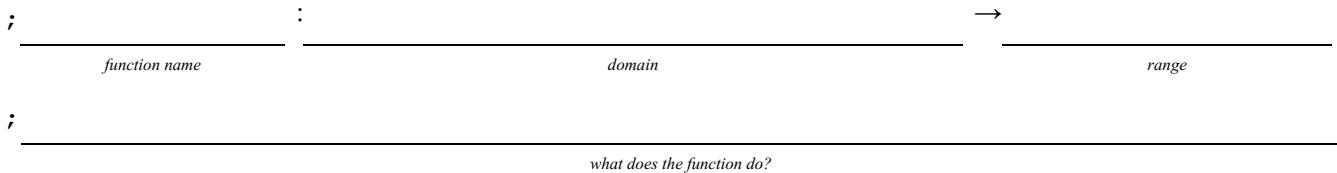


## Word Problem: safe-left?

**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

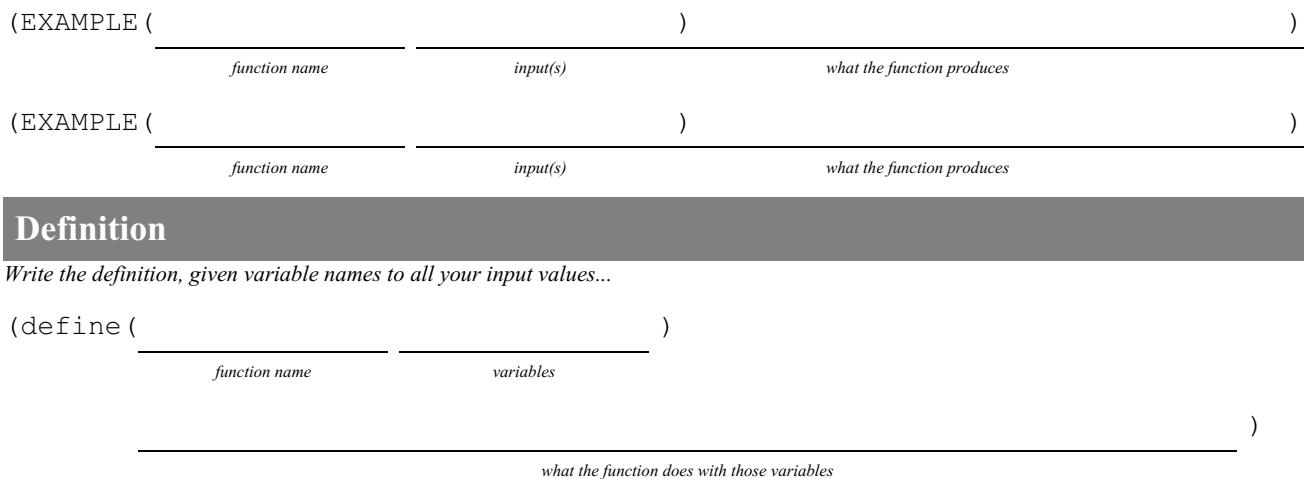
### Contract and Purpose Statement

Every contract has three parts...



### Examples

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

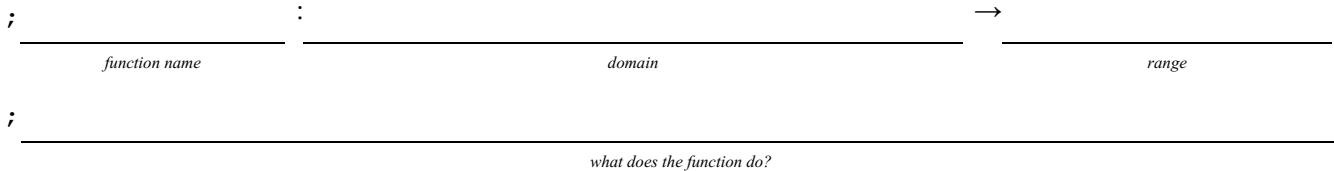


# Word Problem: safe-right?

**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...



## Examples

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

(EXAMPLE ( \_\_\_\_\_ ) )

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

(EXAMPLE ( \_\_\_\_\_ ) )

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

## Definition

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

(define ( \_\_\_\_\_ ) )

*function name*                           *variables*

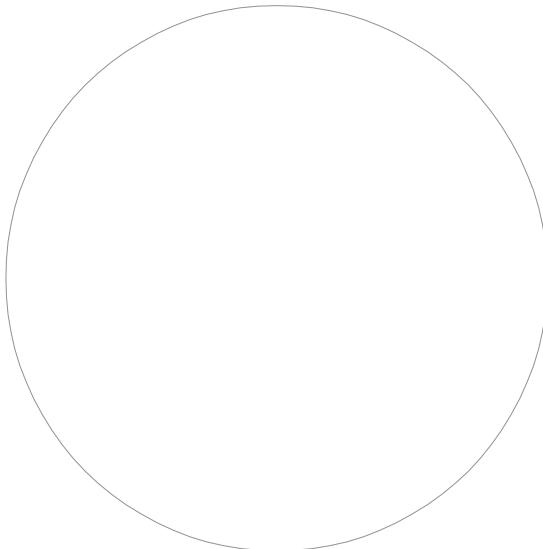
\_\_\_\_\_ )

*what the function does with those variables*

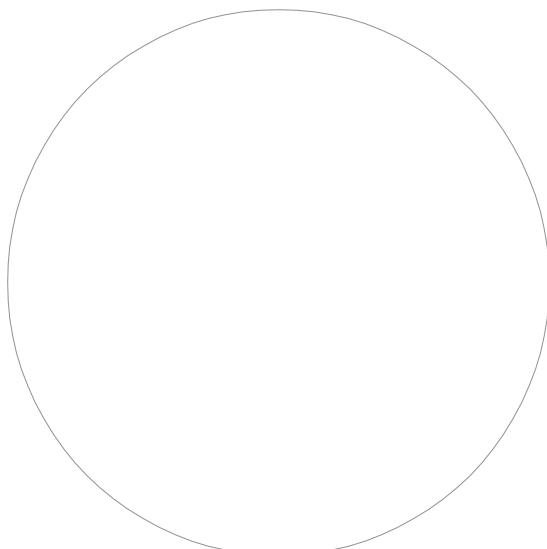
**and / or**

**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.



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



## 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...*

Diagram illustrating the components of a function definition:

The diagram shows a horizontal line divided into three segments by vertical bars. The first segment is labeled *function name*, the second is labeled *domain*, and the third is labeled *range*. An arrow points from the *range* segment to the right. Below the line, the text *what does the function do?* is centered.

## Examples

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

The diagram illustrates function application using two examples. Each example consists of a call to a function, its input, and the resulting output.

**EXAMPLE 1:**

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_                          )  
function name                input(s)

*what the function produces*

**EXAMPLE 2:**

(EXAMPLE ( \_\_\_\_\_ )  
\_\_\_\_\_                          )  
function name                input(s)

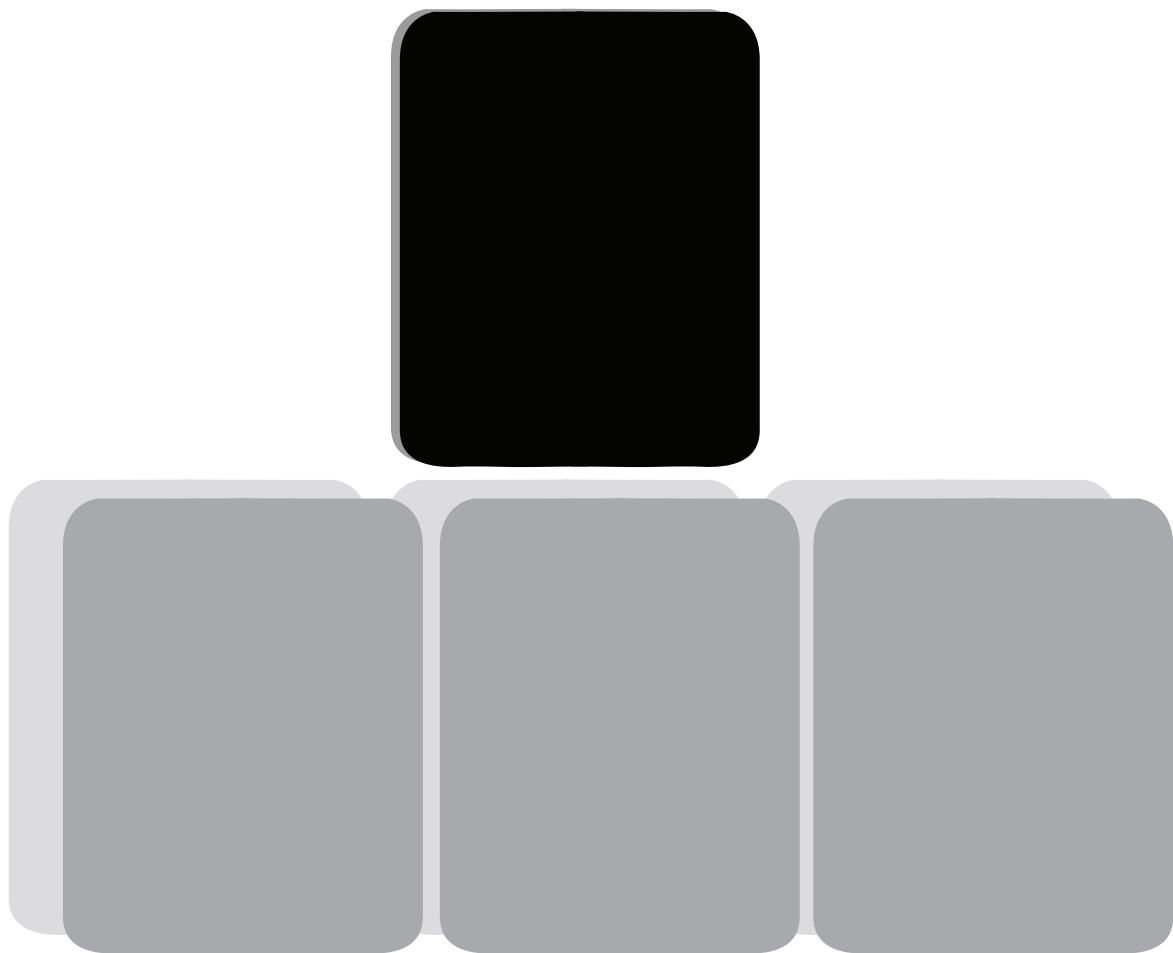
*what the function produces*

## Definition

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

```
(define (function name variables)  
  what the function does with those variables)
```

# 07 Conditional Branching

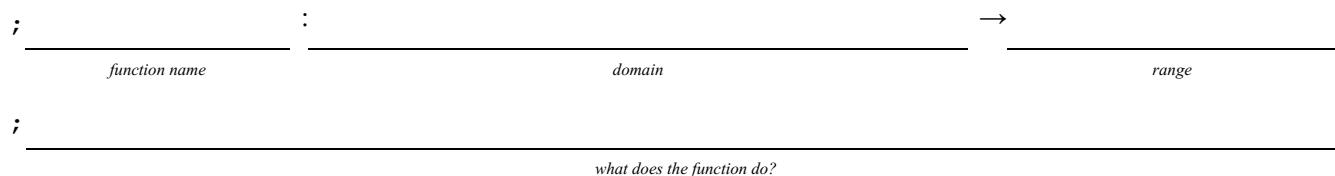


## Word Problem: cost

**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...*



## Examples

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

(EXAMPLE (	<u>cost</u>	"cheese"	)	)
	<i>function name</i>	<i>input(s)</i>	<i>what the function produces</i>	
(EXAMPLE (		)	)	)
	<i>function name</i>	<i>input(s)</i>	<i>what the function produces</i>	
(EXAMPLE (		)	)	)
	<i>function name</i>	<i>input(s)</i>	<i>what the function produces</i>	
(EXAMPLE (		)	)	)
	<i>function name</i>	<i>input(s)</i>	<i>what the function produces</i>	

## Definition

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

```
(define (function name variables)  
  (begin
```

# 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...

;  
\_\_\_\_\_ :  
function name \_\_\_\_\_ → \_\_\_\_\_  
domain \_\_\_\_\_ range  
;  
\_\_\_\_\_  
what does the function do?

## Examples

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

(EXAMPLE ( update-player      320    "up"      )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
function name        input(s)        what the function produces  
  
(EXAMPLE ( update-player      100    "up"      )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
function name        input(s)        what the function produces  
  
(EXAMPLE ( \_\_\_\_\_                  )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
function name        input(s)        what the function produces  
  
(EXAMPLE ( \_\_\_\_\_                  )  
\_\_\_\_\_                  \_\_\_\_\_                  \_\_\_\_\_  
function name        input(s)        what the function produces

## Definition

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

(define ( \_\_\_\_\_ )  
\_\_\_\_\_                  \_\_\_\_\_  
function name        variables  
(  
\_\_\_\_\_  
[  
\_\_\_\_\_ ]  
[  
\_\_\_\_\_ ]  
[ **else**:  
\_\_\_\_\_ ] ) )

# 08 Collision Detection

# collision



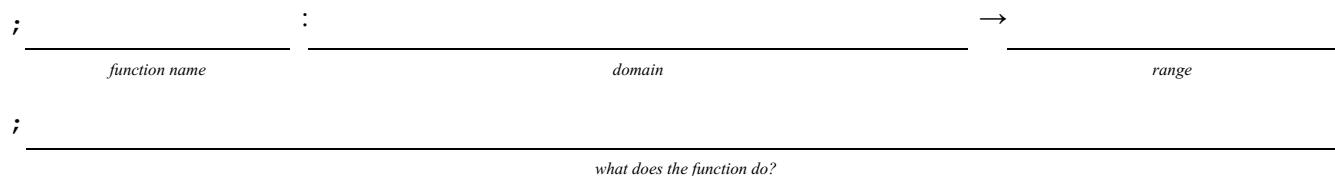
distance

## 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...*



## 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 (	<u>line-length</u>	2 8	)	(- 8 2)	)
	<i>function name</i>	<i>input(s)</i>		<i>what the function produces</i>	

## Definition

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

(define ( *function name* *variables* )

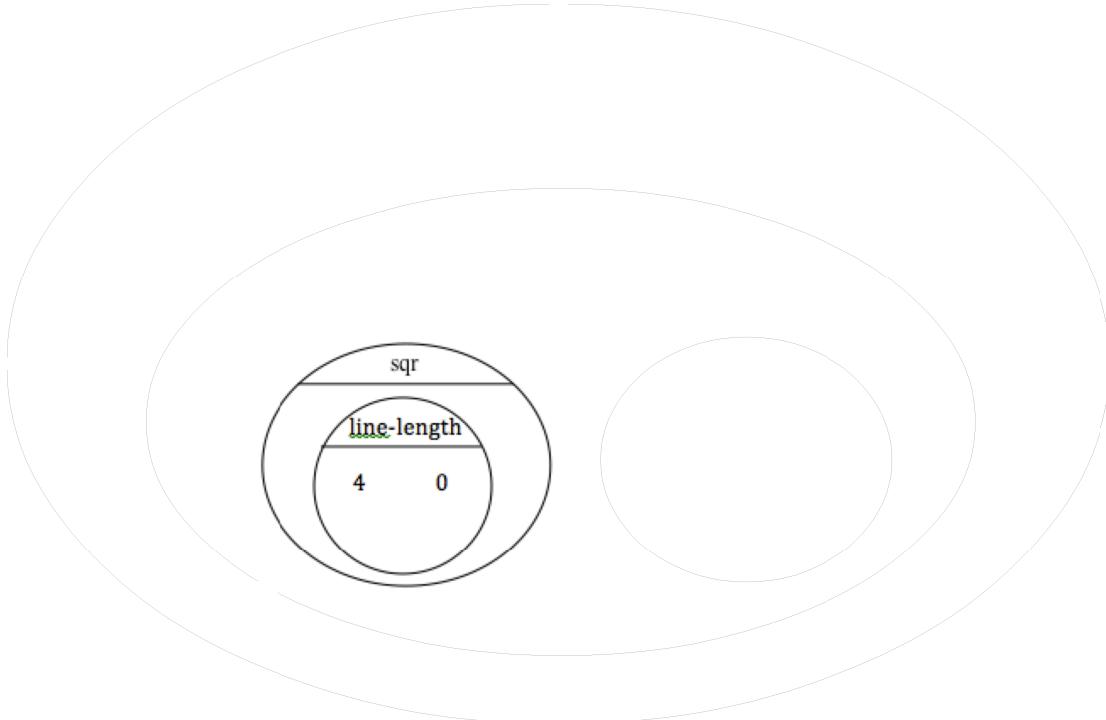
```
(cond  
  _____  
  [  
    _____]  
  [ _____]))
```

## The Distance Formula (an example)

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

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

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  $(x1,y1)$  &  $(y1,y2)$ :

## 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...*

## Examples

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

## Definition

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

```
(define (function name variables)  
  what the function does with those variables)
```

## Word Problem: collide?

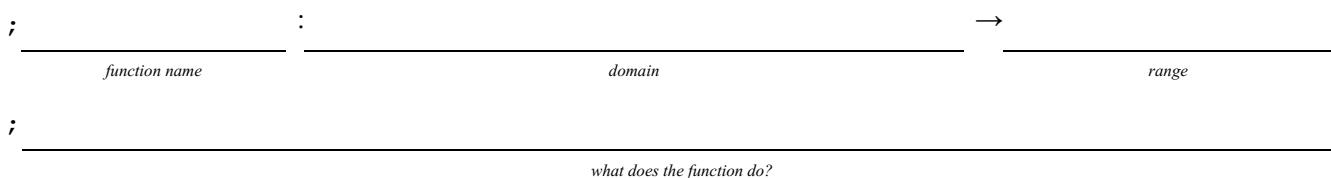
**Directions:** Write a function `collide?`, which takes FOUR inputs:

- $px$ : 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?*

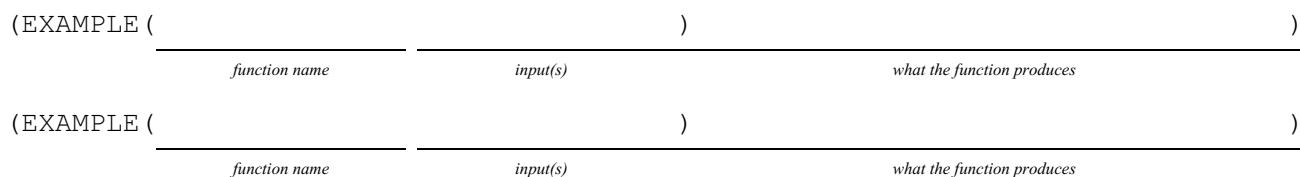
# **Contract and Purpose Statement**

*Every contract has three parts...*



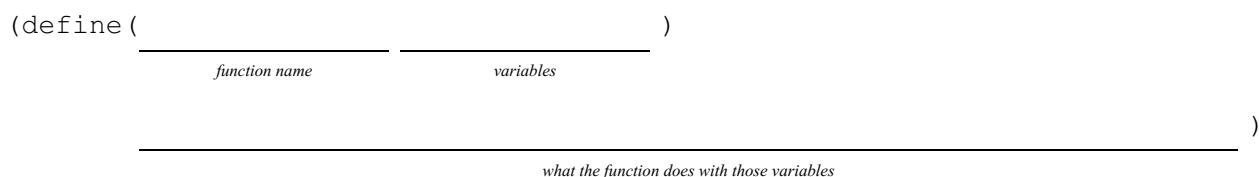
## Examples

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



## Definition

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





## 09 Presentation Preparation



# Lesson 9

Catchy Intro:

---

---

---

Name, Age, Grade:

---

Game Title:

---

---

---

---

Back Story:

---

---

---

---

Characters:

---

---

---

Explain a piece of your code:

---

---

---

---



## 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...*

;

:

→

---

*function name*

*domain*

*range*

;

*what does the function do?*

## Examples

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

(EXAMPLE (	<u>red-shape</u>	"circle"	)	(circle 50 "solid" "red")	)
	<i>function name</i>	<i>input(s)</i>		<i>what the function produces</i>	
(EXAMPLE (			)		)
	<i>function name</i>	<i>input(s)</i>		<i>what the function produces</i>	
(EXAMPLE (			)		)
	<i>function name</i>	<i>input(s)</i>		<i>what the function produces</i>	
(EXAMPLE (			)		)
	<i>function name</i>	<i>input(s)</i>		<i>what the function produces</i>	

## Definition

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

```
(define ( function name variables )  
  (cond  
    [ expression 1 (circle 50 "solid" "red") ]  
    [ expression 2 ]  
    [ expression 3 ]  
    [ expression 4 ]  
    [ else: expression 5 ] ))
```

## Translating into Algebra

### Value Definitions

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

### Function Definitions

Racket Code	Algebra
(define (area length width) (* length width))	$\text{area}(\text{length}, \text{width}) = \text{length} * \text{width}$
(define (circle-area radius) (* pi (sqr radius)))	
(define (distance x1 y1 x2 y2) (sqrt (+ (sqr (- x1 x2)) (sqr (- y1 y2)))))	

# 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 : \_\_\_\_\_ -> \_\_\_\_\_  
name Domain Range  
;  
What does the function do?

## II. Give Examples

Write an example of your function for some sample inputs

D( 1 ) = \_\_\_\_\_  
Use the function here What should the function produce?  
  
D( 2 ) = \_\_\_\_\_  
Use the function here What should the function produce?  
  
D(    ) = \_\_\_\_\_  
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.

D(    ) = \_\_\_\_\_

# 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:

; \_\_\_\_\_ : \_\_\_\_\_ -> \_\_\_\_\_  
; 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?
-----------------------	-----------------------------------

### **III. Definition**

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

# 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:

; \_\_\_\_\_ : \_\_\_\_\_  $\rightarrow$  \_\_\_\_\_  
; 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:**

; \_\_\_\_\_ : \_\_\_\_\_  $\rightarrow$  \_\_\_\_\_  
; name Domain Range  
; \_\_\_\_\_

## **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:**

; \_\_\_\_\_ : \_\_\_\_\_  $\rightarrow$  \_\_\_\_\_  
; name Domain Range  
; \_\_\_\_\_

## **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

## Contracts