

WELCOME TO

TECHNOVATI N

Week 5 - October 22



MICHIGAN STATE UNIVERSITY

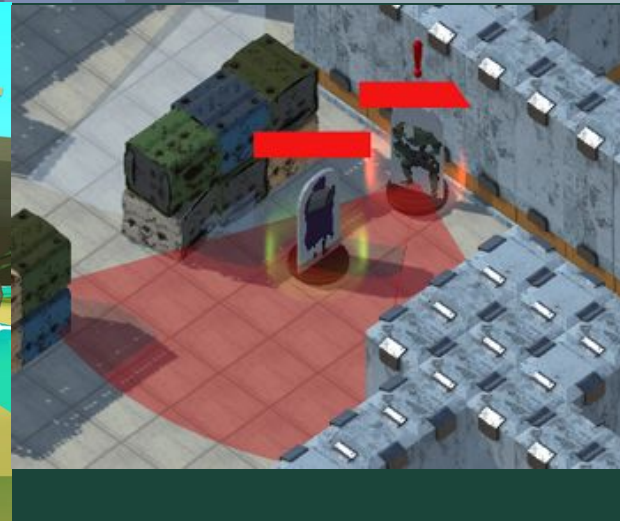
Agenda

- Spotlight
- Lesson:
 - For Loops Review
 - Functions
- Coding challenges
- Standup
- Temperature Check

Spotlight

Video Games!

- To show you some of the cool things you can do with code Jonathon is going to demonstrate some of the games he's made
- You'll have the chance later to play some of the games if you want to!

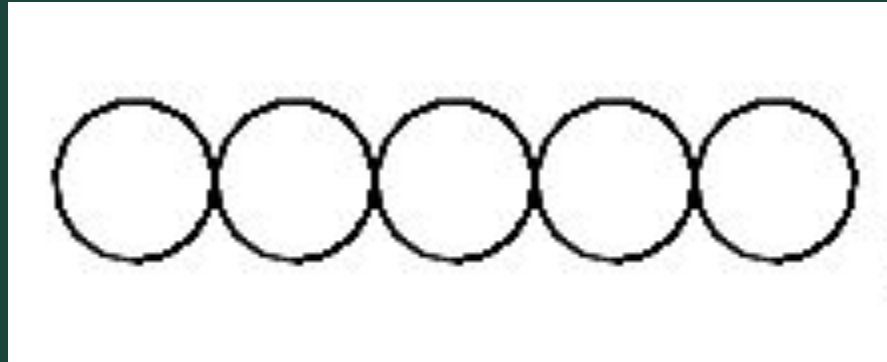


Aaron Solon • Antonio Vasquez • Emma Christensen • Lia Fry
Xiao Ma • Benjamin Person • Jonathon Moore • Sadeem Goji
Dominic Vasquez • Gaby Bendtsen • gamedev.msu.edu

Introducing Loops!

```
1 Tracy, repeat this code 5 times!  
2   circle(20)  
3   penup()  
4   forward(40)  
5   pendown()
```

For loops are used to repeat code a fixed number of times.



Introducing Loops!

For loops help us by:

- shortening our code
- making it easy to alter our code

```
1 circle(20)
2 penup()
3 forward(40)
4 pendown()
5 circle(20)
6 penup()
7 forward(40)
8 pendown()
9 circle(20)
10 penup()
11 forward(40)
12 pendown()
13 circle(20)
14 penup()
15 forward(40)
16 pendown()
17 circle(20)
18 penup()
19 forward(40)
```

19 lines to 5
lines!

```
1 Tracy, repeat this code 5 times!
2   circle(20)
3   penup()
4   forward(40)
5   pendown()
```

Introducing Loops!

```
1 circle(20)
2 penup()
3 forward(40)
4 pendown()
5 circle(20)
6 penup()
7 forward(40)
8 pendown()
9 circle(20)
10 penup()
11 forward(40)
12 pendown()
13 circle(20)
14 penup()
15 forward(40)
16 pendown()
17 circle(20)
18 penup()
19 forward(40)
```

For loops help us by:

- shortening our code
- making it easy to alter our code

Change
radius to 50
pixels

```
1 Tracy, repeat this code 5 times!
2   circle(20)
3   penup()
4   forward(40)
5   pendown()
```

Writing For Loops

`for i in range (amount of times to repeat):`
Commands to repeat go here (indented!)

```
1 Tracy, repeat this code 5 times!  
2     circle(20)  
3     penup()  
4     forward(40)  
5     pendown()
```

Write
loop

```
1 for i in range(5):  
2     circle(20)  
3     penup()  
4     forward(40)  
5     pendown()
```

What is a Function?

Functions are a way to group a set of commands so they can be called with one line of code.

Functions help us teach Tracy new commands using the ones she already knows!

Why Use Functions?

Functions help us by:

- Shortening our code
- Making our code reusable
- Making our code more readable

Defining a Function

```
def function_name_here():  
    function commands here (indented!)
```

Function Name: my_~~function~~

Defining a Function Cont.

```
11 # Draw two circles next to each other
12 for i in range (2):
13     pendown()
14     circle(50)
15     penup()
16     forward(100)
```

```
def function_name_here():
    function commands here
```

Define
function

```
11 # Draw two circles next to each other
12 def draw_two_circles():
13     for i in range (2):
14         pendown()
15         circle(50)
16         penup()
17         forward(100)
```

Calling a Function

```
5 speed(5)
6
7 # This function draws two circles next to each other
8 def draw_two_circles():
9     for i in range (2):
10         pendown()
11         circle(50)
12         penup()
13         forward(100)
14
15 # Move to bottom left of circle group at position (-50,-100)
16 penup()
17 setposition(-50,-100)
18
19 draw_two_circles()
20
21 # Move to top of circle row at position (-50, 0)
22 setposition(-50,0)
23
24 draw_two_circles()
```

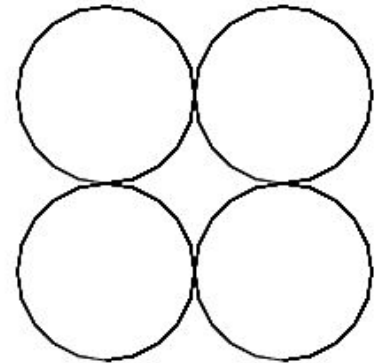
To call a function:

function_name()

*Reminder! Functions must be defined **before** they are called.

Calling a Function

```
5 speed(5)
6
7 # This function draws two circles next to each other
8 def draw_two_circles():
9     for i in range (2):
10         pendown()
11         circle(50)
12         penup()
13         forward(100)
14
15 # Move to bottom left of circle group at position (-50,-100)
16 penup()
17 setposition(-50,-100)
18
19 draw_two_circles()
20
21 # Move to top of circle row at position (-50, 0)
22 setposition(-50,0)
23
24 draw_two_circles()
```



Coding Time

- Let's use today to work on the exercises we haven't been able to finish!
- Work at your own pace! Ask questions!
- If you are all caught up through (Unit 4 Section 2), explore the weekly challenges or the Sandbox in CodeHS

Ready, Set, CODE!

Command	What does it do?
<code>name = value</code>	Saves the value in the variable
<code>input("prompt")</code>	Prints prompt and waits for user input
<code>int(...)</code> , <code>float(...)</code>	Converts a value to a number (int or float)
<code>for i in range(number)</code>	Initialize a loop
<code>def function_name():</code>	Declares a function
<code>function_name()</code>	Calls a function

Command	What does it do?
<code>color("color name")</code>	Changes Tracy's trail color
<code>pensize(number)</code>	Changes Tracy's trail thickness
<code>begin_fill()</code>	Starts tracking closed shapes
<code>end_fill()</code>	Fills & stops tracking closed shapes
<code>setposition(x, y)</code>	Moves Tracy to the input coordinates
<code>speed(number)</code>	Sets how fast Tracy executes commands
<code>name = value</code>	Saves the value in the variable
<code>input("prompt")</code>	Prints prompt and waits for user input
<code>int(...), float(...)</code>	Converts a value to a number (int or float)

Command	What does it do?
<code>forward(<i>distance</i>)</code>	Moves Tracy forward a specified <i>distance</i>
<code>circle(<i>radius</i>)</code>	Draws a circle with a specified <i>radius</i>
<code>backward(<i>distance</i>)</code>	Moves Tracy backward a specified <i>distance</i>
<code>penup()</code>	Stops Tracy from leaving a trail
<code>pendown()</code>	Has Tracy start drawing a trail
<code>left(<i>num</i>)</code>	Turns Tracy <i>num</i> degrees to the left
<code>right(<i>num</i>)</code>	Turns Tracy <i>num</i> degrees to the right

Standup

- What was an exercise you worked on today?
- What is something you were successful at?
- What was a challenge you had while coding?

Logistics: Temperature Check

[Temperature Check](#)